Fall 12-18-2014

Maximum Energy Subsampling: A General Scheme For Multi-resolution Image Representation And Analysis

Yanjun Zhao

MAXIMUM ENERGY SUBSAMPLING: A GENERAL SCHEME FOR MULTI-RESOLUTION IMAGE REPRESENTATION AND ANALYSIS

by

YANJUN ZHAO

Under the Direction of Saeid Belkasim, PhD

ABSTRACT

Image descriptors play an important role in image representation and analysis. Multi-resolution image descriptors can effectively characterize complex images and extract their hidden information.

Wavelets descriptors have been widely used in multi-resolution image analysis. However, making the wavelets transform shift and rotation invariant produces redundancy and requires complex matching processes. As to other multi-resolution descriptors, they usually depend on other theories or information, such as filtering function, prior-domain knowledge, etc.; that not only increases the computation complexity, but also generates errors.

We propose a novel multi-resolution scheme that is capable of transforming any kind of image descriptor into its multi-resolution structure with high computation accuracy and efficiency. Our multi-resolution scheme is based on sub-sampling an image into an odd-even image tree. Through applying image descriptors to the odd-even image tree, we get the relative multi-resolution image descriptors. Multi-resolution analysis is based on downsampling expansion with maximum energy extraction followed by upsampling reconstruction. Since the maximum energy usually retained in the lowest frequency coefficients; we do maximum energy extraction through keeping the lowest P coefficients from each resolution level.

Our multi-resolution scheme can analyze images recursively and effectively without introducing artifacts or changes to the original images, produce multi-resolution representations, obtain higher resolution images only using information from lower resolutions, compress data, filter noise, extract effective image features and be implemented in parallel processing.

INDEX WORDS: Multi-resolution image descriptors, Multi-resolution analysis, Maximum energy extraction, Feature extraction, Image compression, Image Denoising

MAXIMUM ENERGY SUBSAMPLING: A GENERAL SCHEME FOR MULTI-RESOLUTION IMAGE REPRESENTATION AND ANALYSIS

by

YANJUN ZHAO

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in the College of Arts and Sciences

Georgia State University

2014

MAXIMUM ENERGY SUBSAMPLING: A GENERAL SCHEME FOR MULTI-RESOLUTION IMAGE REPRESENTATION AND ANALYSIS

by

YANJUN ZHAO

Committee Chair: Saeid Belkasim

Committee: Raj Sunderraman

Ying Zhu

Hendricus Van Der Holst

Marina Arav

Electronic Version Approved:

Office of Graduate Studies

College of Arts and Sciences

Georgia State University

December 2014

DEDICATION

I dedicate this dissertation to my God and Lord.

Give thanks to the LORD, for he is good; his love endures forever.

-------- Psalm 107:1

ACKNOWLEDGEMENTS

The monumental task of a doctoral degree can only be undertaken with the consistent support and encouragement of many individuals. First and foremost, I must thank my supervisor Prof. Saeid Belkasim for directing my research all these years. His instruction, patience, and auspice have made me the researcher I am today. Working with Prof. Saeid Belkasim has been a great honor, and I look forward to our continued collaboration. Many thanks go to my committee members Prof. Raj Sunderraman, Prof. Ying Zhu, Prof. Hendricus Van Der Holst and Prof. Marina Arav for their valuable, constructive feedback on my research. Special thanks to Prof. Ben Miller from Departments of English and Communication, for asking me to join his interdisciplinary research group "Digging into Human Rights Violations" where I greatly open my minds. I am grateful to the faculty who taught me at Georgia State University; in their classes, I learned a lot. Especially for Prof. Sushil Prasad, he gave me plenty of instructions on algorithm design and analysis. Gratitude to Ms.Tammie Dudley, for her steadfast supports and boundless efforts for my PhD studying.

I am grateful to my team workers, Sanghoon Lee, Semra Kul, Mary Hudachek-Buswell, Janani Krishnamani, Ayush Shrestha, Nicholas Subtirelu, Jin Zhao, Kristopher Kyle, Jennifer Olive, for their great supports for my research.

Lastly, I wish to thank my family and friends. It is only through their love, care, and supports that I am able to finish this research marathon.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

1.1 Background and Motivations

A digital image consists of a collection of pixels that record light intensities at specific locations in a grid matrix of elements known as pixels [35, 161, 167-169]. Image descriptor is a set of numbers to describe a set of given image features. Usually, an image descriptor is in the form of a vector; it quantifies image to agree with human intuition or task-specific requirements. Image descriptors play an important role in image analysis [1-4, 23, 28, 30, 35, 38-39].

There are mainly two groups of image descriptors: contour-based ones, extracting image feature only from the contour of an image and region-based ones, extracting image feature from the whole image region [16, 50-88]. Under each group, the different methods can be further divided into structural and global methods; this classification is based on whether the image is represented as a whole or represented by segments [49].

Current image descriptors mainly face four challenges:

- Lower computation Complexity: With the development of internet, the on-line image processing requires higher computation efficiency. However, the current image descriptors with good identify and invariance properties usually have high computation complexity while the ones with low computation complexity usually have poor identify and invariance properties.
- Lower storage requirements: The amount of images that need to store increase hugely, however, the storage space is limited. Therefore, each image has to be reduced to a compact feature vector to fit the limited storage space. Currently, image descriptors which can represent image with acceptable quality usually require more image information and do not have good compactness ability.

- Robustness: Noise is usually unavoidable in the real world image database. However, the reduction of noise sensitivity is the common problem that most of the current image descriptors have to face, especially for the image descriptors that have high computation efficiency, such as contour-based image descriptors.
- Flexibility: As to the same image, different applications usually requires different kinds of representation, therefore, we need the image descriptors that have strong flexibility to describe and analyze an image in different ways based on different requirements. Structure image descriptor allows the partition of an image, which provides a way for different representation [18, 49]. However the structure methods usually depend on prior-knowledge and have high computation complexity; they are sensitive to noise and fail to represent the global features.

Multi-resolution image descriptors can represent and analysis an image in different resolution levels. The high resolutions provide detail information while the low resolutions show general views. This hierarchical structure identifies important features of an image among the unimportant ones; therefore we can efficiently compact images and meet the low storage requirements. Furthermore, the values of noise data are usually different with general image data. Through performing statistical computation of the coefficients from the same or different resolution levels, we can filter the noise and increase the robustness. And also, the nature of multi-resolution representation greatly improves the flexibility to represent an image.

Therefore, multi-resolution image descriptors are good candidates to deal with these challenges. Unfortunately, the current multi-resolution descriptors suffer different kinds of limitations. Here we give a review of these descriptors.

(1) Wavelets

The fundamental idea behind wavelets is to analyze data based on scale (resolution). The wavelets are scaled and translated copies ("daughter wavelets") of a finite-length or fast-decaying oscillating waveform ("mother wavelet"). A wavelet transform is the representation of a function by wavelets. Multi-resolution analysis based on wavelet transform allows the introduction of the concept of details among successive levels of scale (resolution) [14-15, 27, 31-33, 41-42, 89-94, 154]. Currently, wavelets descriptor is the most popular multi-resolution image descriptor. However, there are mainly three disadvantages of wavelets descriptor [94]:

- Shift sensitivity: Wavelets are seriously disadvantaged by the shift sensitivity that arises from down samplers. Shift sensitivity is an undesirable property since it implies that wavelets coefficients fail to distinguish among input-signal shifts.
- Poor directionality: An m-Dimensional transform (m>1) suffers poor directionality when the transform coefficients reveal only a few feature orientations in the spatial domain. Wavelets have poor directional selectivity for diagonal features, since the wavelet filters are separable and real.
- Absence of phase information: Phase information is useful in many areas of image processing such as image compression, power measurement [95]. However most wavelets cannot provide the local phase information.

(2) Image pyramid

An image pyramid is a collection of decreasing resolutions arranged in an image of a pyramid: The base of the pyramid contains a high-resolution while the apex contains a low-resolution; when moving up the pyramid, both size and resolution decrease [35, 96-97, 170].

There are mainly two types of intimately related image pyramids: approximation pyramid and prediction residual pyramid [35, 170]. An approximation pyramid is generated by a reducing function that is used to smooth the image with an appropriate approximation filter and downsample the smoothed image. A classical approximation pyramid is Gaussian pyramid. A prediction residual pyramid is obtained by forming the difference between adjacent levels in an approximation pyramid, where an expansion function that includes upsampler and interpolation filter is performed between representations at adjacent resolution levels to make the computation of pixel be different. A classical prediction residual pyramid is Laplacian pyramid.

Image pyramids has simple data structure and easy to be understood. However, the construction image pyramid depends on filters which not only increases its computation complexity but also change the original information of the image.

(3) Scale space method

Scale-space method is a formal method to process image structures at different scales, through representing an image as a one-parameter family of smoothed images; the scale-space representation is parameterized by the size of the smoothing kernel that is used for suppressing fine-scale structures [10, 37, 98-102].

A typical approach for choosing a type of scale space representation is to establish a set of scale-space axioms to describe basic properties of the desired scale-space representation and make the representation useful in practical applications [100]. A set of standard scale space axioms leads to linear Gaussian scale-space. The corresponding scale-space framework encompasses a theory for Gaussian derivative operators that can be used as a basis for expressing a large group of visual operations to process visual information [100].

Curvature scale space (CSS) and intersection points map (IPM) are two popular scale-space methods. They both use a Gaussian kernel to progressively smooth a curve to a varying bandwidth [17, 38, 40]. Although CSS is robust to noises, changes in scale and orientation of an image makes high computation complexity. IMP is faster than CSS; however IMP is failure to handle occulted contours.

Scale space methods can get abundant information of a shape with different scales; their main advantages are high robustness to noise and good coherence with human perception. However, there are three main disadvantages of scale space methods:

- Loss original information: although the low-pass filter in scale space methods can generate the multi-scale representation of an image, the corresponding smoothing process definably changes the original image, which is not allowed in many real applications.
- Depending on prior-domain knowledge: the setting of the scale parameters (e.g. the width of Gaussian filter) usually depends on the prior-domain knowledge; the prior-domain knowledge cannot be always available.
- High computation complexity: in the real applications, the scale-based methods usually suffer complex matching process; the filtering processing also adds more computation tasks.

(4) Fractal geometry

A fractal is a property of self-similarity which means that an object is composed of subunits and sub-sub-units on multiple levels that statistically resemble the structure of the whole object [43, 47-48]. An important characteristic of a fractal object is that its measured metric properties are a function of the scale of measurement.

A fractal object has a fractional dimension which is a statistical quantity that indicates how completely a fractal appears to fill space, when an image is zoomed down to finer and finer scales. The fractional dimension assumes that the length of an object depends on the resolution or the scale at which this object is measured. Through calculating and combining different forms of fractal dimension, the irregularity of complex shapes can be quantified [45].

The main attraction of fractal geometry comes from its ability to describe and characterize the complexity of images, especially for their texture composition [44, 46, 103]. However, this method suffers some limitations [103-104]:

- Computing the fractional dimension: Although there are many computation methods, they suffer different kinds of limitations. It is hard to know whether the observed differences in computed fractional dimension values is a result of true differences in image texture or a result of certain arbitrary decisions made during the estimation process.
- Describing an object with multi-fractal behavior by a single fractal value: Multi-fractal analysis is a response to this limit, which allows computing a spectrum or a set of fractal dimensions. However, many algorithms exist to evaluate this spectrum and numerical differences between the methods appear. How to choose a suitable algorithm is an open problem.

(5) Binary partition tree

A binary partition tree is a hierarchical region-based representation of an image that can be implemented by the initial partition [34, 36]. The leaves of the tree represent regions that belong to the initial partition; the root node represents the entire image support; the remaining nodes of the tree represent regions that are obtained by merging the regions represented by the

two children of the node. This tree represents a set of regions at different scales. Large regions are close to the root whereas small details are close to the lower levels.

The creation of a binary partition tree relies on two main notions: the merging criterion that defines the similarity of neighboring regions and the order in which regions are going to be merged and the region model that specifies how to represent the regions and how to compute the model of the union of two regions [34, 36].

A binary partition tree reflects the similarity between neighboring regions; it has two main disadvantages:

- Inapplicable for contour-based image: The binary partition tree is only for region-based image. However, there are many image descriptors that are contour-based; at that case, the binary partition tree becomes inapplicable.
- Defining a suitable merging criterion: The creation of a binary partition tree is based on a merging criterion; however, there are many kinds of merging criterion, how to define a suitable merging criterion based on the real applications and requirements becomes a question.

(6) Adaptive grid resolution

The adaptive grid resolution (AGR) represents a 2D image through adaptively modeling different portions of the image at different resolutions [38-39]. In the AGR, a square grid that is just big enough to cover the entire shape is overlaid on an image. A resolution of the grid cells varies from one portion to another according to the content of the shape portion: On the borders or the detail portion of the shape, the higher resolution, i.e, the smaller grid cells, are applied while in the coarse regions of the shape, lower resolution, i.e. the bigger grid cells, are applied.

Computing AGR requires a quad-tree decomposition of an image. This decomposition is based on successive subdivision the bitmap into four equal-size quadrants. If a bitmap-quadrant does not consist entirely of "1s" or entirely of "0s" (i.e. the shape "partially covers" the quadrant), it is recursively subdivided into smaller and smaller quadrants until reaching the bitmap-quadrants, possibly 1×1 pixels in size, consisting entirely of "1s" or entirely of "0s", or the termination condition of the recursion that we predefined [38-39].

AGR is amenable to use existed multidimensional index structures and support efficient similarity retrieval. However, there are several limitations of AGR:

- Inapplicable for contour-based image: AGR is only for region-based image. However, there are also many images that are contour-based; at that case, AGR becomes impracticable.
- Computation redundancy: Since we use a square grid to cover a shape which is usually not a square, the square grid has to include some redundant areas that do not belong to the original shape to ensure the entire coverage.
- Diversity of the quad-tree: Different decomposition start position of a shape produce different quad-tree. Therefore, how to choose a suitable decomposition start position of a shape becomes a problem.
- Sensitivity to noise: When we generate the quad-tree, we require that the bitmap-quadrant should consist entirely of "1s" or entirely of "0s". Therefore, a small amount noises will greatly change our generation results.

Although these multi-resolutions theories have their own advantages, they depend on other theories (except wavelets descriptor), which not only increases the computation complexity but also generates errors. Image pyramid and scale space method depend on filtering function;

the filtering function changes the original information of the image and adds more computation tasks. Fractal geometry depends on the computation of fractional dimension; the current computation methods suffer different kinds of limitations which usually generate computation errors and redundancy. Binary partition tree and adaptive grid resolution are based on region segmentation; segmentation itself remains a largely open problem, that further complicates the situation. Although wavelets are independent, their shift sensitivity and poor directivity usually brings out complex matching process, which make them impracticable in many real applications.

1.2 Multi-resolution Image Representation and Analysis

We propose a novel multi-resolution image representation and analysis scheme that can be adapted to any kind of descriptors for generating their multi-resolution image structures with good invariance and computation properties. Our multi-resolution scheme is based on transforming each image into an odd-even image tree. Through applying image descriptors to this odd-even image tree, we get the relative multi-resolution image descriptors. Multi-resolution analysis is based on downsampling expansion with maximum energy extraction followed by upsampling reconstruction. Since the maximum energy usually retained in the lowest frequency coefficients; we do maximum energy extraction through keeping the lowest P coefficients from each resolution level. Our scheme can analyze images recursively and effectively without introducing artifacts or changes to the original image, produce different kinds of multi-resolution representations, obtain higher resolution images only using information from lower resolution levels, compress data, filter noise, extract effective image features and be implemented in parallel processing.

Since image descriptors can be mainly classified as contour-based and region-based descriptors; our multi-resolution scheme also has two formats: boundary-based scheme which

generates boundary-based multi-resolution descriptors and region-based scheme which generates region-based multi-resolution descriptors.

Applications of our multi-resolution scheme are explored in the areas of feature extraction, image compression and denoising.

Analyzing with a large number of variables generally requires a large amount of memory and computation power; it may make a classification algorithm over fit the training sample and generalize poorly to new samples. Feature extraction plays important roles in image processing and pattern recognition. However, the current feature extraction algorithms are either restricted by assumptions or hard to be implemented or suffering high computation complexity.

Exploring features (coefficients) in different resolutions reveals the dominant features (coefficients) in comparison to the redundant ones. We propose a novel feature extraction algorithm based on our multi-resolution scheme. Our algorithm is free from assumption, robust to noise, adaptive to any kind of image descriptor, and can be implemented by parallel processing.

The goal of image compression is to reduce irrelevance and redundancy of the image data for storing or transmitting it efficiently while preserving its quality for the given application. Since noise is usually unavoidable in the real world image data. If we can combine image compression with image denoising, that is compacting an image though reducing the noise, it will be very prospective.

Currently, the most popular image descriptor that can do image compression and denoising at the same time is wavelets; however wavelets suffer high computation complexity and poor invariance property.

We propose a novel image compression and denoising algorithm based on our multi-resolution scheme. Our algorithm has outstanding performance in compressing data and filtering noise; it is free from block artifacts, adaptive to any kind of image descriptor and ideal for parallel computing. Our algorithm outperforms wavelets in terms of computation accuracy and efficiency.

We arrange the structure of our dissertation as follows: In chapter 1, we give an introduction. In chapter 2, we propose our odd-even image tree. In chapter 3, we propose our novel multi-resolution image representation and analysis scheme based on our odd-even image tree. In chapter 4, we propose a novel feature extraction algorithm based on our multi-resolution scheme. In chapter 5, we propose a novel image compression and denoising algorithm based on our multi-resolution scheme. In chapter 6, we give the conclusions and future works.

2 ODD-EVEN IMAGE TREE

Our multi-resolution image representation and analysis scheme is based on transforming an image into an odd-even image tree. In this chapter, we propose this odd-even image tree. Since our multi-resolution scheme has two formats: boundary-based and region-based; our odd-even image tree also has two formats: an odd-even image boundary tree relative to a boundary-based multi-resolution scheme and an odd-even image region tree relative to a region-based multi-resolution scheme.

2.1 Odd-even Image Boundary Tree

Our odd-even image boundary tree is based on dyadic ally splitting pixels of an image boundary into an odd pixels set and an even pixels set.

2.1.1 Traditional Image Boundary Representation

A boundary is a collection of pixels that encloses a region. A digital boundary in xy-plane can be represented by N pixels. Starting at an arbitrary point (x_0, y_0), the coordinate pairs (x_0, y_0), (x_1, y_1), … , (x_k, y_k), …, (x_{N-1}, y_{N-1}) are traversed in the counter-clockwise direction, which is shown in Figure 2.1 [35].

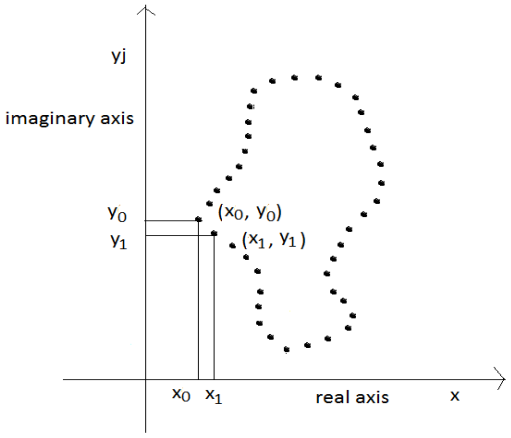

Figure 2.1 Image Boundary

These coordinates can be expressed in the form $x(k) = x_k$ and $y(k) = y_k$[35]. Therefore the boundary can be represented as a sequence of coordinates:

$$S(k) = [x(k), y(k)] \tag{2.1}$$

$$where \;\; k = 0, 1, 2, \dots, N-1.$$

Moreover, each coordinate pair can be treated as a complex number [35]:

$$S(k) = x(k) + jy(k) \tag{2.2}$$

$$where\ k = 0,1,2, \dots, N-1; j = \sqrt{-1}\ .$$

The x-axis represents the real part and the y-axis represents the part of a sequence of complex numbers. It reduces a 2-D problem to a 1-D problem.

2.1.2 Odd-even Image Boundary Tree

We dyadic ally split the pixels of an image boundary $S(k), k = 0, 1, 2, , \dots, N-1$ into two sets: a set of odd pixels that contains the odd pixels of the boundary and a set of even pixels that contains the even pixels of the boundary [109]. The structure of this procedure is represented by an odd-even image boundary tree shown in Figure 2.2.

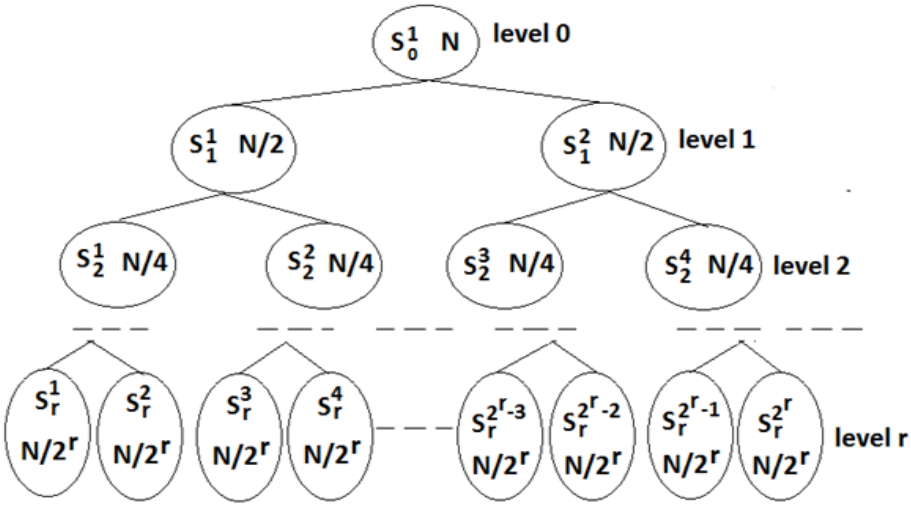

Figure 2.2 Odd-even Image Boundary Tree
S: boundary; N: number of pixels in the original boundary

In this tree, the resolution level is inversely proportional to the height of the tree: the highest resolution occurs at the root whereas the lowest resolution occurs at the leaves. Each other node has a left and right child of lower resolution and a parent of higher resolution. The number of pixels in a node is half of its parent.

The multi-resolution boundary $S_r^m(k_r^m)$ at node m of tree level r is defined [109]:

$$S_r^m(k_r^m) = x(k_r^m) + jy(k_r^m) \tag{2.3}$$

Where $k_r^m = 0,1,2,\dots,N_r - 1;\ N_r = \frac{N}{2^r}$*; N is the number of pixels in the original boundary*

$$m = 1,2,3,\dots,2^r; r = 0,1,2,\dots; j = \sqrt{-1}\ .$$

At level zero $r = 0$, there is only a boundary S_0^1. It represents the original boundary [109]:

$$S_0^1(k_0^1) = x(k_0^1) + jy(k_0^1) \tag{2.4}$$

Where $k_0^1 = 0,1,2,\dots,N_0 - 1\ .$

At level $r(r > 0)$, there are 2^{r-1} pairs of odd-even boundaries $S_r^{m_o}$ and $S_r^{m_e}$[109]:

$$\begin{cases} S_r^{m_o}\left(k_r^{m_o}\right) = S_{r-1}^m(2i) \\ S_r^{m_e}\left(k_r^{m_e}\right) = S_{r-1}^m(2i+1) \end{cases} \tag{2.5}$$

Where $k_r^{m_o} = k_r^{m_e} = i = 0,1,2,3,\dots,N_r - 1;$

$$m_o = 1,3,5,\dots,2^r - 1; m_e = m_o + 1;\ m = 1,2,3,\dots,2^{r-1}.$$

Each pair of odd-even boundaries $S_r^{m_o}$and $S_r^{m_e}$ is generated by its parent boundary $S_{r-1}^m$from higher resolution level.

The hierarchical representation of the odd-even image boundary tree allows us to transfer different kinds of boundary-based image descriptors into their multi-resolution structures for multi-resolution analysis. This tree structure is also ideal for parallel processing that can greatly improve the computation efficiency.

2.2 Odd-even Image Region Tree

In digital image processing, a $K_1\ by\ K_2$ image is usually represented as a $K_1\ by\ K_2$ matrix $H(k_1, k_2)$ where $k_1 = 0, 1, 2, \dots, K_1 - 1$ and $k_2 = 0, 1, 2, \dots, K_2 - 1$.

An odd-even image region tree is constructed by dyadically dividing an image $H(k_1, k_2)$ where $k_1 = 0, 1, 2, \dots, K_1 - 1$ and $k_2 = 0, 1, 2, \dots, K_2 - 1$ into four sets: a set of odd-odd pixels that contains pixels of odd rows and odd columns; a set of odd-even pixels that contains pixels of odd rows and even columns, a set of even-odd pixels that contains pixels of even rows and odd columns, and a set of even-even pixels that contains pixels of even rows and even columns [110]. This image tree is shown in Figure 2.3.

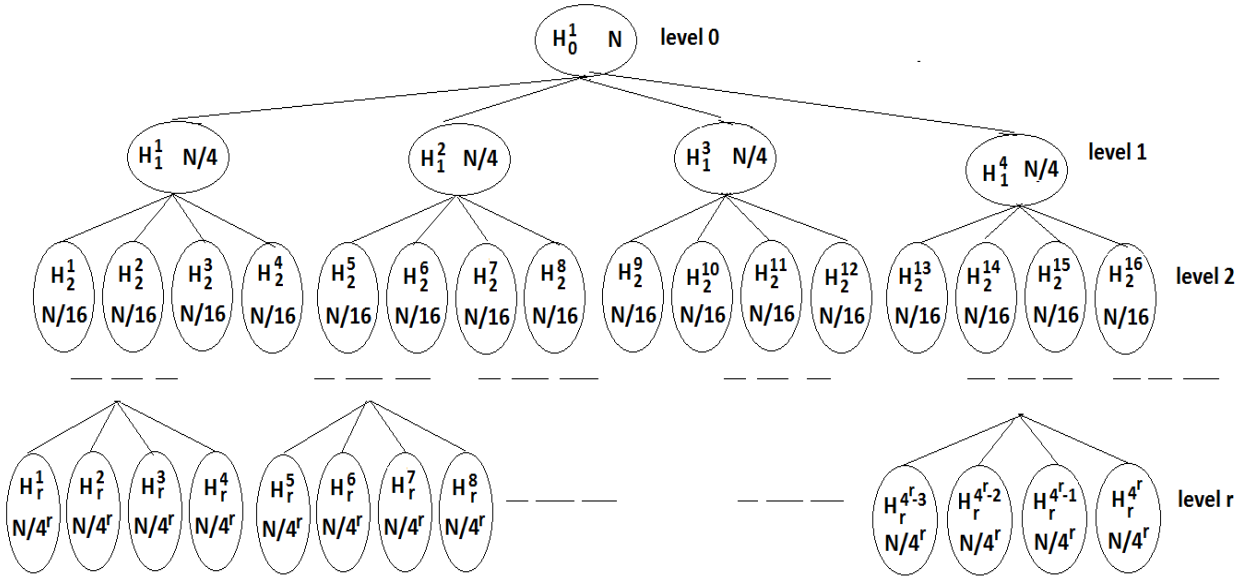

Figure 2.3 Odd-even Image Region Tree
H : Image matrix ; N: the number of pixels in the original image matrix $N = K_1 \times K_2$

In this tree, the resolution level is inversely proportional to the height of the tree: the highest resolution occurs at the root that can lossless represent the original image whereas the lowest resolution occurs at the leaves; each other node has four children of lower resolution in terms of odd-odd, odd-even, even-odd and even-even image and a parent node of higher resolution. The numbers of rows and columns of an image at a node are respectively a half of its parent node; the size of an image at a node is a quarter of its parent node [110].

The multi-resolution image at node m of tree level r is represented as $H_r^m((k_1)_r^m, (k_2)_r^m)$ $where (k_1)_r^m = 0, 1, 2, \dots (K_1)_r - 1;\ (K_1)_r = \frac{K_1}{2^r}; (k_2)_r^m = 0, 1, 2, \dots (K_2)_r - 1;\ (K_2)_r = \frac{K_2}{2^r};\ m = 1, 2, 3, \dots, 4^{\mathrm{r}}$ and $r = 0,1,2, \dots$.

At level $\mathrm{r} = 0$, there is only one image H_0^1. It represents the original image as follows [110]:

$$H_0^1((k_1)_0^1, (k_2)_0^1) = H(k_1, k_2) \tag{2.6}$$

$$Where\ (k_1)_0^1 = k_1 = 0, 1, 2, \dots, K_1 - 1\ ;\ (k_2)_0^1 = k_2 = 0, 1, 2, \dots, K_2 - 1.$$

At level $\mathrm{r} > 0$, there are $4^{\mathrm{r}-1}$ groups of odd-odd, odd-even, even-odd and even-even image $H_r^{m_{oo}}\ H_r^{m_{oe}} H_r^{m_{eo}}\ H_r^{m_{ee}}$ as follows[110]:

$$\begin{cases} H_r^{m_{oo}}((k_1)_r^{m_{oo}}, (k_2)_r^{m_{oo}}) = H_{r-1}^m((2i_1), (2i_2)) \\ H_r^{m_{oe}}((k_1)_r^{m_{oe}}, (k_2)_r^{m_{oe}}) = H_{r-1}^m((2i_1), (2i_2 + 1)) \\ H_r^{m_{eo}}((k_1)_r^{m_{eo}}, (k_2)_r^{m_{eo}}) = H_{r-1}^m((2i_1 + 1), (2i_2)) \\ H_r^{m_{ee}}\left((k_1)_r^{m_{ee}}, (k_2)_r^{m_{ee}}\right) = H_{r-1}^m((2i_1 + 1), (2i_2 + 1)) \end{cases} \tag{2.7}$$

$$Where\ (k_1)_r^{m_{oo}} = (k_1)_r^{m_{oe}} = (k_1)_r^{m_{eo}} = (k_1)_r^{m_{ee}} = i_1 = 0, 1, 2, \dots, \frac{K_1}{2^r} - 1;$$

$$(k_2)_r^{m_{oo}} = (k_2)_r^{m_{oe}} = (k_2)_r^{m_{eo}} = (k_2)_r^{m_{ee}} = i_2 = 0, 1, 2\ \dots, \frac{K_2}{2^r} - 1;$$

$$m_{oo} = 1,5, 9, 13, 17, \dots, 4^r - 3; m_{oe} = m_{oo} + 1;$$

$$m_{eo} = m_{oo} + 2; m_{ee} = m_{oo} + 3;$$

$$m = 1, 2, \dots, 4^{r-1},\ m\ is\ ID\ of\ a\ node\ at\ level\ r.$$

Each group of odd-odd, odd-even, even-odd and even-even image $H_r^{m_{oo}}$ $H_r^{m_{oe}}$ $H_r^{m_{eo}}$ and $H_r^{m_{ee}}$ is generated by its parent image H_{r-1}^{m} from higher resolution level.

The hierarchical representation of the odd-even image region tree allows us to transfer different kinds of region-based image descriptors into their multi-resolution structures for multi-resolution analysis. This tree structure is also ideal for parallel processing that can greatly improve the computation efficiency.

Summary

In this chapter, we introduce our odd-even image tree that is the foundation of our multi-resolution image representation and analysis scheme. Our odd-even image tree has two formats: an odd-even image boundary tree relative to a boundary-based multi-resolution scheme and an odd-even image region tree relative to a region-based multi-resolution scheme. This tree structure is ideal for parallel processing that can greatly improve the computation efficiency.

3 MULTI-RESOLUTION IMAGE REPRESENTATION AND ANALYSIS SCHEME

Our multi-resolution image representation and analysis scheme is based on transforming each image into an odd-even image tree.

In this chapter, we propose our novel multi-resolution image representation and analysis scheme. Through applying different kinds of image descriptors to our odd-even image tree, we get different kinds of multi-resolution image descriptors. The multi-resolution analysis is achieved based on downsampling expansion with maximum energy extraction followed by upsampling reconstruction. Since the maximum energy usually retained in the lowest frequency coefficients; we do maximum energy sampling through keeping the lowest P coefficients from each resolution level. The basic structure of this scheme is shown in Figure 3.1.

Our multi-resolution scheme can analyze images recursively and effectively without introducing artifacts or changes to the original images, provide multi-resolution representations, obtain higher resolution images only using information from lower resolutions, compress data, filter noise, extract effective image features and be implemented in parallel processing.

Since there are mainly two groups of image descriptors: boundary-based and region-based; our multi-resolution scheme also have two formats: a boundary-based scheme that generates boundary-based multi-resolution descriptors and a region-based scheme that generates region-based multi-resolution descriptors.

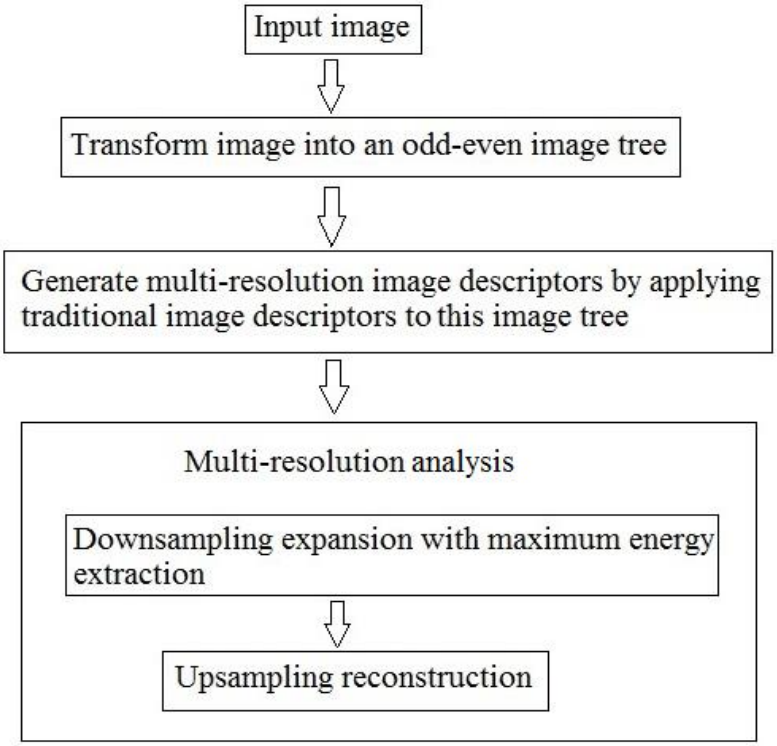

Figure 3.1 Multi-resolution Scheme

3.1 Boundary-based Multi-resolution Scheme

In this section, we propose our novel boundary-based multi-resolution scheme: we apply boundary-based image descriptors to the odd-even image boundary tree, to get the relative multi-resolution boundary-based image descriptors. The multi-resolution image analysis is achieved based on downsampling expansion with maximum energy extraction followed by upsampling reconstruction. Since the maximum energy usually retained in the lowest frequency coefficients; we do maximum energy extraction through keeping the lowest P coefficients from each resolution level. The basic structure of this multi-resolution scheme is shown in Figure 3.2.

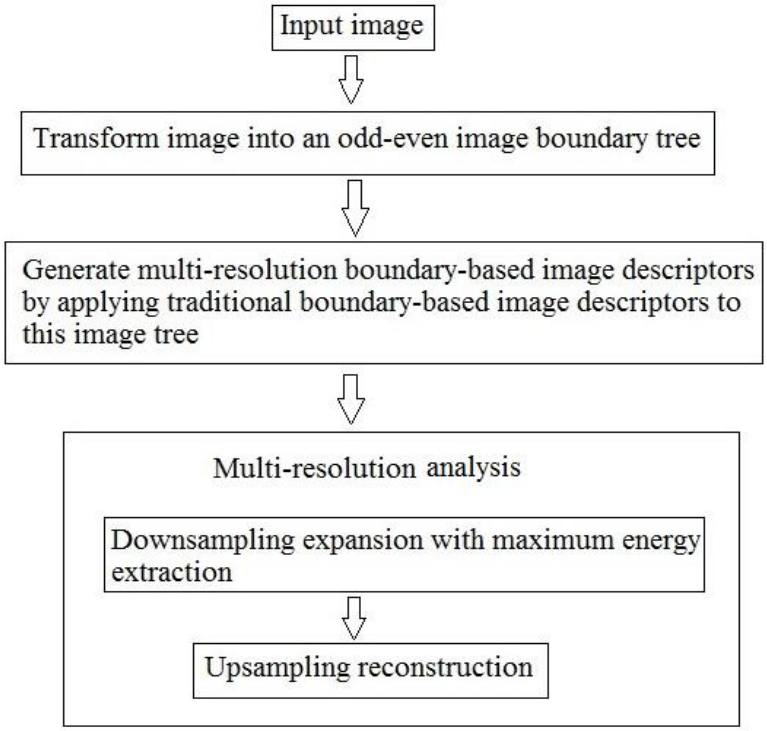

Figure 3.2 Boundary-based Multi-resolution Scheme

3.1.1 Boundary-based Multi-resolution Image Descriptors

Through applying different kinds of boundary –based image descriptors to the odd-even image boundary tree, we can get different kinds of boundary-based multi-resolution image descriptors. Since there are many boundary-based image descriptors; we take Fourier descriptors, one of the most widely used boundary-based image descriptors [19, 20-22, 25, 29, 35], as an example to introduce our boundary-based multi-resolution scheme; the procedure for other boundary-based image descriptors is the same except replacing Fourier descriptors with other image descriptors.

(1) Traditional Fourier descriptors

Fourier descriptors, is a promising boundary-based approach [25, 29, 35]. The key idea of traditional Fourier descriptors is to apply Fourier transform to a periodic representation of contour, which results in an image descriptor set in frequency domain [35].

The discrete Fourier transform of the boundary $S(k)$ is defined as follows [35]:

$$F(u) = \sum_{k=0}^{N-1} S(k) e^{-\frac{j2\pi uk}{N}} \tag{3.1}$$

$$\textit{Where } u = 0, 1, 2, \dots, N-1; j = \sqrt{-1}\,.$$

The complex coefficients $F(u)$ are the Fourier descriptors of the boundary $S(k)$.

The inverse Fourier transform of Fourier descriptors restores boundary $S(k)$ [35]:

$$S'(k) = \frac{1}{N} \sum_{u=0}^{N-1} F(u) e^{\frac{j2\pi uk}{N}} \tag{3.2}$$

$$\textit{Where } k = 0,1,2, \dots, N-1; j = \sqrt{-1}\,.$$

$S'(k)$ is the reconstructed boundary based on the Fourier descriptors with all of the coefficients.

Low-frequency coefficients determine global information while high-frequency coefficients account for fine details.

Fourier descriptors are one of the dominant boundary-based approaches. The compactness and computational efficiency of Fourier descriptors make them very popular. In [20] there is a comparison between Fourier descriptors and autoregressive models in term of image classification; Fourier descriptors are proved to be better. In [21] there is a comparison among chain codes, Fourier descriptors, and geometric moments, in term of image similarity-based retrieval; moments and Fourier descriptors get the best performance. In [22] Fourier descriptors gave better experimental results in image retrieval than CSS-representation. Many applications show that Fourier descriptors outperform other boundary-based techniques [3, 11-12, 19].

However, traditional Fourier descriptors cannot provide multi-resolution representation and analysis, which greatly impacts their applications.

(2) Multi-resolution Fourier descriptors

We apply the traditional Fourier descriptors to the odd-even image boundary tree to generate the multi-resolution Fourier descriptors. The multi-resolution Fourier descriptors F_r^m are defined as follows [109]:

$$F_r^m(u_r^m) = \sum_{k_r^m=0}^{N_r-1} S_r^m(k_r^m) e^{-\frac{j2\pi u_r^m k_r^m}{N_r}} \tag{3.3}$$

Where $u_r^m = 0,1,2,\dots,N_r-1; N_r = \frac{N}{2^r};\ m = 1,2,3,\dots,2^r; r = 0,1,2,\dots;\ j = \sqrt{-1}$.

When $r = 0$ we get the root of the multi-resolution Fourier descriptors [109]:

$$F_0^1(u_0^1) = \sum_{k_0^1=0}^{N_0-1} S_0^1(k_0^1) e^{-\frac{j2\pi u_0^1 k_0^1}{N_0}} \tag{3.4}$$

Where $u_0^1 = 0,1,2,\dots,N_0-1.$

When $r > 0$, we get 2^{r-1} pairs of odd-even Fourier descriptors $F_r^{m_o}$ and $F_r^{m_e}$ [109]:

$$\begin{cases} F_r^{m_o}\left(u_r^{m_o}\right) = \sum_{k_r^{m_o}=0}^{N_r-1} S_r^{m_o}\left(k_r^{m_o}\right) e^{-\frac{j2\pi u_r^{m_o} k_r^{m_o}}{N_r}} \\ F_r^{m_e}\left(u_r^{m_e}\right) = \sum_{k_r^{m_e}=0}^{N_r-1} S_r^{m_e}\left(k_r^{m_e}\right) e^{-\frac{j2\pi u_r^{m_e} k_r^{m_e}}{N_r}} \end{cases} \tag{3.5}$$

Where $m_o = 1,3,5,\dots,2^r-1; m_e = m_o+1;\ u_r^{m_o} = u_r^{m_e} = 0,1,2,\dots,N_r-1$.

Our boundary-based multi-resolution scheme can generate other kinds of boundary-based multi-resolution image descriptors under the same procedure except replacing Fourier descriptors with other kinds of boundary-based image descriptors.

The boundary-based multi-resolution image descriptor can be considered as a coefficients tree; the resolution level is inversely proportional to the height of the tree: the highest resolution occurs at the root whereas the lowest resolution occurs at the leaves. Each other node has a left and right child of lower resolution and a parent of higher resolution. The number of coefficients at a node is half of its parent. Exactly, the coefficients tree has the same structure with the odd-even image boundary tree, except replacing a set of image pixels with a set of coefficients.

Obviously, this procedure to generate the boundary-based multi-resolution image descriptors has a binary tree structure which is ideal for parallel computing.

3.1.2 Boundary-based Multi-resolution Analysis

The boundary-based multi-resolution analysis can be achieved based on downsampling expansion with maximum energy extraction followed by upsampling reconstruction.

Downsampling expansion with maximum energy extraction: the highest resolution image boundary is reconstructed from the root of the coefficients tree, each pair of lower resolution image boundaries is reconstructed respectively through each pair of relative lower resolution coefficients sets. Since the maximum energy is contained within the lowest frequency coefficients; we do maximum energy extraction by keeping the lowest P coefficients, that is computing image only based on the lowest P coefficients.

Upsampling reconstruction: Each odd-even pair of lower resolution reconstructed image boundaries are merged to form a higher resolution parent reconstructed image boundary; this procedure is repeated until we arrive at the root.

We also take Fourier descriptors as an example to illustrate the boundary-based multi-resolution analysis. This multi-resolution analysis procedure is shown in Figure 3.3.

(1) Boundary-based down sampling expansion with maximum energy extraction

The downsampling expanded boundary I_r^m is computed as follows [109]:

$$I_r^m(k_r^m) = \frac{1}{P_r^m} \sum_{u_r^m=0}^{P_r^m-1} F_r^m(u_r^m) e^{\frac{j2\pi u_r^m k_r^m}{P_r^m}} \tag{3.6}$$

Where $m = 1, 2, 3, \dots, 2^r;\ k_r^m = 0,1,2, \dots, N_r - 1;\ 0 < P_r^m \leq N_r;\ j = \sqrt{-1}$.

Since maximum energy is contained within the lowest frequency coefficients; keeping the lowest P coefficients is poised to retain essential information and improves computation efficiency.

When $r = 0$, we get the root of the downsampling expanded boundary as follows[109]:

$$I_0^1(k_0^1) = \frac{1}{P_0^1} \sum_{u_0^1=0}^{P_0^1-1} F_0^1(u_0^1) e^{\frac{j2\pi u_0^1 k_0^1}{P_0^1}} \tag{3.7}$$

Where $k_0^1 = 0,1,2, \dots, N_0 - 1;\ 0 <\ P_0^1 \leq N_0.$

When r > 0, we get 2^{r-1} pairs of odd-even downsampling expanded boundaries $I_r^{m_o}$ and $I_r^{m_e}$ as follows [109]:

$$\begin{cases} I_r^{m_o}\left(k_r^{m_o}\right) = \dfrac{1}{P_r^{m_o}} \displaystyle\sum_{u_r^{m_o}=0}^{P_r^{m_o}-1} F_r^{m_o}\left(u_r^{m_o}\right) e^{\frac{j2\pi u_r^{m_o} k_r^{m_o}}{P_r^{m_o}}} \\ I_r^{m_e}\left(k_r^{m_e}\right) = \dfrac{1}{P_r^{m_e}} \displaystyle\sum_{u_r^{m_e}=0}^{P_r^{m_e}-1} F_r^{m_e}\left(u_r^{m_e}\right) e^{\frac{j2\pi u_r^{m_e} k_r^{m_e}}{P_r^{m_e}}} \end{cases} \tag{3.8}$$

Where $m_o = 1, 3, 5, \dots, 2^r - 1; m_e = m_o + 1;\ k_r^{m_o} = k_r^{m_e} = 0,1,2, \dots, N_r - 1;$

$$0 < P_r^{m_o} \le N_r \; ; 0 < P_r^{m_e} \le N_r$$

The downsampling procedure has a binary tree structure that is ideal for parallel computing.

(2) Boundary-based upsampling reconstruction

When $r = R - 1$ where R is the height of the odd-even image boundary tree, we get 2^{R-1} upsampling reconstructed boundaries through merging 2^{R-1}pairs of odd-even downsampling expanded boundaries $I_R^{m_o}$and $I_R^{m_e}$ from lower resolution level as follows [109]:

$$C_{R-1}^{h}\left(k_{R-1}^{h}\right) = \begin{cases} I_R^{m_o}\left(k_R^{m_o}\right) & \text{when } k_{R-1}^{h} = 2i \\ I_R^{m_e}\left(k_R^{m_e}\right) & \text{when } k_{R-1}^{h} = 2i+1 \end{cases} \tag{3.9}$$

$$\textit{Where } k_R^{m_o} = k_R^{m_e} = i = 0, 1, 2, \ldots, N_R - 1;\; k_{R-1}^{h} = 0, 1, 2, \ldots, N_{R-1} - 1;$$

$$h = 1, 2, 3, \ldots, 2^{R-1};\; m_o = 1, 3, 5, \ldots, 2^R - 1;\; m_e = m_o + 1.$$

We consider these 2^{R-1} upsampling reconstructed boundaries as 2^{R-2} pairs of odd-even upsampling reconstructed boundaries $C_{R-1}^{h_o}$and $C_{R-1}^{h_e}$ where $h_o = 1, 3, 5, \ldots, 2^{R-1} - 1$ and $h_e = h_o + 1$.

When $r < R - 1$, we get 2^{r-1} pairs of odd-even upsampling reconstructed boundaries $C_r^{h_o}$and $C_r^{h_e}$ where $h_o = 1, 3, 5, \ldots, 2^r - 1$ and $h_e = h_o + 1$ by merging 2^rpairs of odd-even upsampling reconstructed boundaries $C_{r+1}^{h'_o}$ and $C_{r+1}^{h'_e}$ from lower resolution level. The merging rule is the same except that we merge $C_{r+1}^{h'_o}$ and $C_{r+1}^{h'_e}$ instead of $I_R^{m_o}$ and $I_R^{m_e}$. This procedure is repeated until arriving at the root where we get C_0^1 with the same size of the original boundary.

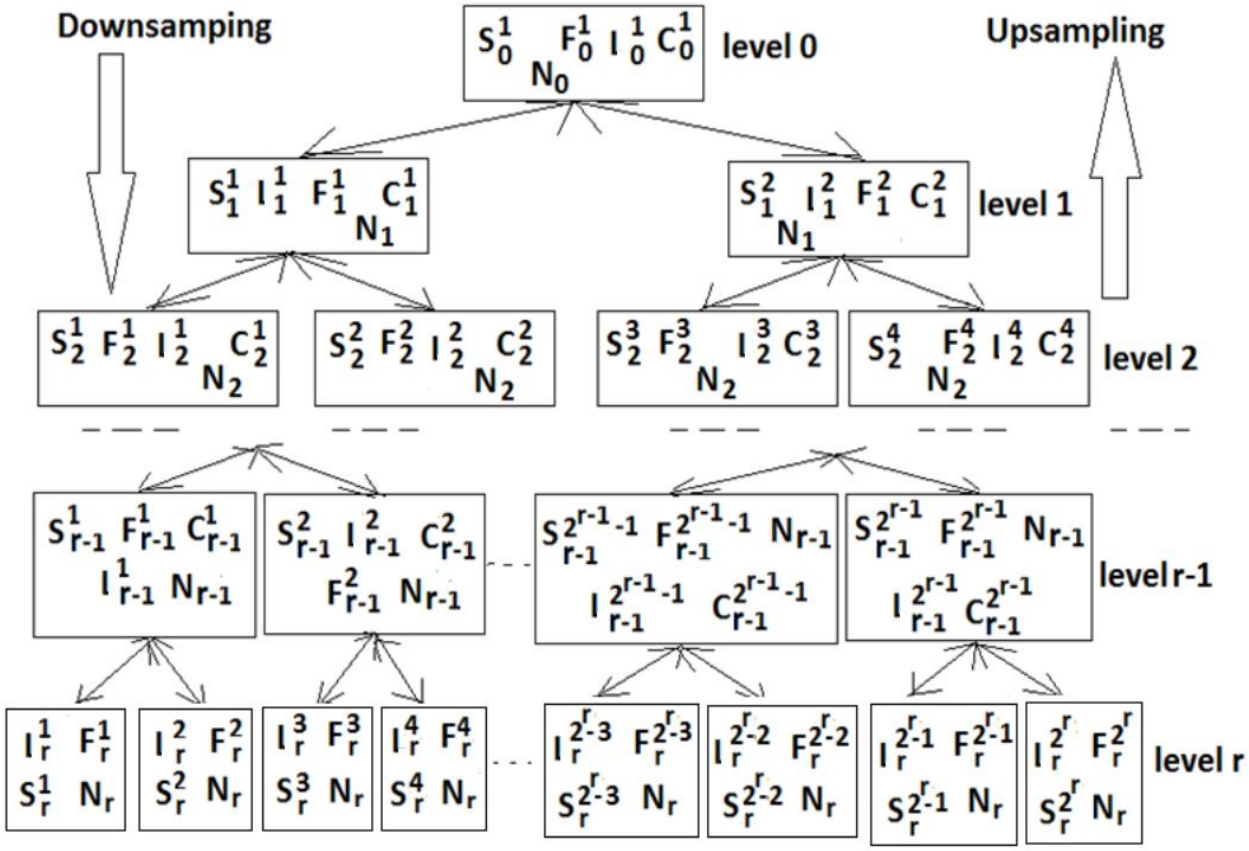

Figure 3.3 Boundary-based Multi-resolution Image Analysis
S: boundary from the odd-even image boundary tree; I: down sampling expanded boundary;
C: upsampling reconstructed boundary; F: multi-resolution Fourier descriptor;
N: number of pixels in a boundary.

We can do boundary-based multi-resolution analysis with other boundary-based multi-resolution image descriptors under the same procedure except replacing multi-resolution Fourier descriptors with other boundary-based multi-resolution image descriptors. The procedures of downsampling and upsampling have tree structures that are ideal for parallel processing.

3.2 Region-based Multi-resolution Scheme

In this section, we propose our novel region-based multi-resolution scheme: we apply the traditional region-based image descriptors to the odd-even image region tree, to get the related region-based multi-resolution image descriptors. The region-based multi-resolution analysis is

achieved based on downsampling expansion with maximum energy extraction followed by upsampling reconstruction. The basic structure of this region-based multi-resolution scheme is shown in Figure 3.4.

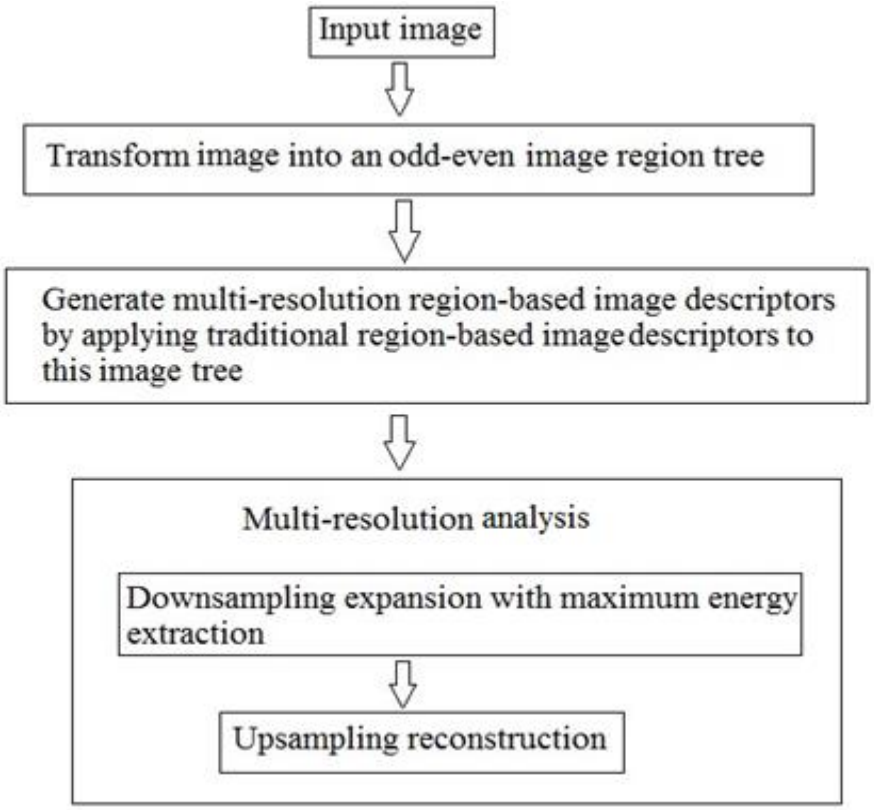

Figure 3.4 Region-based Multi-resolution Scheme

3.2.1 Region-based Multi-resolution Image Descriptors

Through applying different kinds of region–based image descriptors to the odd-even image region tree, we can get different kinds of region-based multi-resolution image descriptors. Since there are many region-based image descriptors; we take Zernike moments, especially Cartesian Zernike moments, a promising region-based image descriptors, as an example to introduce our region-based multi-resolution image descriptors; the procedure for generating other region-based multi-resolution image descriptors is the same except replacing Zernike moments with other region-based descriptors.

(1) Traditional Cartesian Zernike moments

Zernike moments are one of the most prospective region-based image descriptors [6-8, 26, 83, 85, 106, 112]. They are super in terms of feature representation capability [5, 13], invariance to linear transformations and rotation [9, 24], and better image reconstruction [107-108]. One of the major problems of Zernike moments is the incompatibility of the image function in Cartesian coordinates and the Zernike polynomials in polar coordinates. This incompatibility affects the orthogonally condition of Zernike polynomials and may produce genomic errors as well as increasing the computation complexity.

S. Belkasim et al [105] expressed Zernike moments in Cartesian coordinates to mainly avoid this problem. Therefore, we take Cartesian Zernike Moments as an example to demonstrate our region-based multi-resolution image descriptors.

Zernike moments are defined in terms of a set of orthogonal functions with simple rotation properties known as Zernike polynomials. Zernike polynomials can be expressed in Cartesian coordinates as follows [105]:

$$V_{nL}(x,y) = R_{nL}(x,y)\left((x^2+y^2)^{\frac{-L}{2}}\sum_{f=0}^{m_d}(-1)^f\binom{L}{2f}x^{L-2f}y^{2f} + j\left((x^2+y^2)^{\frac{-L}{2}}\sum_{f=0}^{m_q}(-1)^f\binom{L}{2f+1}x^{L-2f-1}y^{2f+1}\right)\right) \quad (3.10)$$

Where $j=\sqrt{-1}$ *; n: order, non-negative integer;*

L: repetition, non negative integer subject to the constraints $n-L$ *is even and* $L \leq n$.

$R_{nL}(x,y)$*: a real valued Zernike polynomial defined:*

$$R_{nL}(x,y) = \sum_{s=0}^{(n-L)/2} (-1)^s \frac{(n-s)!}{s!\left(\frac{n+L}{2}-s\right)!\left(\frac{n-L}{2}-s\right)!} (x^2+y^2)^{\frac{n-2s}{2}} \quad (3.11)$$

m_d and m_q are defined with respect to L:

when L is even $m_d = \frac{L}{2}$, $m_q = \frac{L-2}{2}$*; when L is odd* $m_d = \frac{L-1}{2}$, $m_q = \frac{L-1}{2}$.

In digital image processing, a K_1 *by* K_2 image is usually represented as a K_1 *by* K_2 matrix $H(k_1, k_2)$where $k_1 = 0, 1, 2, \dots, K_1 - 1$ and $k_2 = 0, 1, 2, \dots, K_2 - 1$. We use Cartesian Zernike moments to represent an image $H(k_1, k_2)$ as follows [105]:

$$A_{nL} = \frac{n+1}{\pi |H(k_1,k_2)|} \sum_{k_1=0}^{K_1-1} \sum_{k_2=0}^{K_2-1} H(k_1,k_2) V_{nL}^*(k_1,k_2) \quad (3.12)$$

The symbol * denotes the complex conjugate of V_{nL} .

Since repetition L is decided by order n; given n a maximum value Q, we can have a set of Cartesian Zernike moments to describe an image as follows:

$$A_{nL} = \{A_{00}, A_{20}, A_{22}, A_{31}, A_{33}, \dots, A_{Q\,(Q-2)}, A_{QQ}\} \quad (3.13)$$

Where $n = 0, 1, \dots, Q;$ L*: non negative integer,* $n - L$ *is even and* $L \leq n$.

An image can be reconstructed by Cartesian Zernike moments as follows:

$$H'(k_1,k_2) = \sum_{n=0}^{Q} \sum_{L=0}^{n} A_{nL} V_{nL}(k_1,k_2) \quad (3.14)$$

Where $n = 0, 1, \dots, Q;$ L*: non negative integer* $n - L$ *is even and* $L \leq n$

Cartesian Zernike moments do not transfer the image into the frequency domain. The essential information of an image is not only contained by the coefficients with the lower order. Since it is difficult to define the most effective Cartesian Zernike coefficients in a concise way; how to choose effective Cartesian Zernike coefficients is still an open problem. In chapter four, we will propose a novel coefficients extraction algorithm to deal with this problem.

(2) Multi-resolution Cartesian Zernike moments

We apply the Cartesian Zernike moments to the odd-even image region tree to generate the multi-resolution Cartesian Zernike moments. The multi-resolution Cartesian Zernike moments $(A_{nL})_r^m$ are defined as follows [110]:

$$(A_{nL})_r^m = \frac{n+1}{\pi |H_r^m((k_1)_r^m,(k_2)_r^m)|} \sum_{(k_1)_r^m=0}^{(K_1)_r-1} \sum_{(k_2)_r^m=0}^{(K_2)_r-1} H_r^m((k_1)_r^m,(k_2)_r^m) \times V_{nL}^*((k_1)_r^m,(k_2)_r^m) \quad (3.15)$$

Where $H_r^m((k_1)_r^m,(k_2)_r^m)$ *is the multi-resolution image at node m of tree level r*

$$|H_r^m((k_1)_r^m,(k_2)_r^m)| = (K_1)_r \times (K_2)_r = (K_1/2^r) \times (K_2/2^r) \quad (3.16)$$

$V_{nL}((k_1)_r^m,(k_2)_r^m)$ is the multi-resolution Cartesian Zernike moments polynomials

$$V_{nL}((k_1)_r^m,(k_2)_r^m)$$

$$= R_{nL}((k_1)_r^m,(k_2)_r^m)\Bigg(\left(((k_1)_r^m)^2+((k_2)_r^m)^2\right)^{\frac{-L}{2}} \sum_{f=0}^{m_d} (-1)^f \binom{L}{2f} ((k_1)_r^m)^{L-2f} ((k_2)_r^m)^{2f}$$

$$+ j\left(((k_1)_r^m)^2\right.$$

$$\left. + ((k_2)_r^m)^2\right)^{\frac{-L}{2}} \sum_{f=0}^{m_q} (-1)^f \binom{L}{2f+1} ((k_1)_r^m)^{L-2f-1} ((k_2)_r^m)^{2f+1}\Bigg) \quad (3.17)$$

$j = \sqrt{-1}$ *; n: order, non-negative integer;*

L: repetition, non negative integer subject to the constraints $n - L$ *is even and* $L \leq n$.

$$R_{nL}((k_1)_r^m,(k_2)_r^m) = \sum_{s=0}^{(n-L)/2} (-1)^s \frac{(n-s)!}{s!\left(\frac{n+L}{2}-s\right)!\left(\frac{n-L}{2}-s\right)!} \left(((k_1)_r^m)^2+((k_2)_r^m)^2\right)^{\frac{n-2s}{2}} \quad (3.18)$$

m_d and m_q are defined with respect to L:

when L is even $m_d = \frac{L}{2}$ $m_q = \frac{L-2}{2}$; when L is odd $m_d = \frac{L-1}{2}$; $m_q = \frac{L-1}{2}$;

$$m = 1, 2, 3, \dots, 4^r;\ r = 0,1,2, \dots .$$

When $r = 0$ we get the root of the multi-resolution Cartesian Zernike moments $(A_{nL})_0^1$:

$$(A_{nL})_0^1 = \frac{n+1}{\pi |H_0^1((k_1)_0^1, (k_2)_0^1)|} \sum_{(k_1)_0^1=0}^{(K_1)_0-1} \sum_{(k_2)_0^1=0}^{(K_2)_0-1} H_0^1((k_1)_0^1, (k_2)_0^1) \times V_{nL}^*((k_1)_0^1, (k_2)_0^1) \quad (3.19)$$

When r > 0, we get 4^{r-1} groups of odd-odd, odd-even, even-odd and even-even Cartesian Zernike moments $(A_{nL})_r^{m_{oo}}$, $(A_{nL})_r^{m_{oe}}$, $(A_{nL})_r^{m_{eo}}$ and $(A_{nL})_r^{m_{ee}}$:

$$\begin{cases} (A_{nL})_r^{m_{oo}} = \frac{n+1}{\pi |H_r^{m_{oo}}((k_1)_r^{m_{oo}}, (k_2)_r^{m_{oo}})|} \sum_{(k_1)_r^{m_{oo}}=0}^{(K_1)_r-1} \sum_{(k_2)_r^{m_{oo}}=0}^{(K_2)_r-1} H_r^{m_{oo}}((k_1)_r^{m_{oo}}, (k_2)_r^{m_{oo}}) \times V_{nL}^*\left((k_1)_r^{m_{oo}}, (k_2)_r^{m_{oo}}\right) \\ (A_{nL})_r^{m_{oe}} = \frac{n+1}{\pi |H_r^{m_{oe}}((k_1)_r^{m_{oe}}, (k_2)_r^{m_{oe}})|} \sum_{(k_1)_r^{m_{oe}}=0}^{(K_1)_r-1} \sum_{(k_2)_r^{m_{oe}}=0}^{(K_2)_r-1} H_r^{m_{oe}}((k_1)_r^{m_{oe}}, (k_2)_r^{m_{oe}}) \times V_{nL}^*\left((k_1)_r^{m_{oe}}, (k_2)_r^{m_{oe}}\right) \\ (A_{nL})_r^{m_{eo}} = \frac{n+1}{\pi |H_r^{m_{eo}}((k_1)_r^{m_{eo}}, (k_2)_r^{m_{eo}})|} \sum_{(k_1)_r^{m_{eo}}=0}^{(K_1)_r-1} \sum_{(k_2)_r^{m_{eo}}=0}^{(K_2)_r-1} H_r^{m_{eo}}((k_1)_r^{m_{eo}}, (k_2)_r^{m_{eo}}) \times V_{nL}^*\left((k_1)_r^{m_{eo}}, (k_2)_r^{m_{eo}}\right) \\ (A_{nL})_r^{m_{ee}} = \frac{n+1}{\pi |H_r^{m_{ee}}((k_1)_r^{m_{ee}}, (k_2)_r^{m_{ee}})|} \sum_{(k_1)_r^{m_{ee}}=0}^{(K_1)_r-1} \sum_{(k_2)_r^{m_{ee}}=0}^{(K_2)_r-1} H_r^{m_{ee}}((k_1)_r^{m_{ee}}, (k_2)_r^{m_{ee}}) \times V_{nL}^*\left((k_1)_r^{m_{ee}}, (k_2)_r^{m_{ee}}\right) \end{cases} \quad (3.20)$$

Where $m_{oo} = 1,5,9,13,17,\ldots,4^r - 3; m_{oe} = m_{oo} + 1;\ m_{eo} = m_{oo} + 2; m_{ee} = m_{oo} + 3$.

We can apply our region-based multi-resolution scheme to generate other region-based multi-resolution image descriptors under the same procedure except replacing Cartesian Zernike moments with other kinds of region-based image descriptors.

The region-based multi-resolution image descriptor can be considered as a coefficients tree; the resolution level is inversely proportional to the height of the tree: the highest resolution occurs at the root whereas the lowest resolution occurs at the leaves. Each other node has four children of lower resolution and a parent of higher resolution. The number of coefficients at a node is a quarter of its parent. The coefficients tree has the same structure with the odd-even image region tree, except replacing a set of image pixels with a set of coefficients.

Obviously, this procedure to generate the region-based multi-resolution image descriptors has a quarter tree structure which is ideal for parallel computing.

3.2.2 Region-based Multi-resolution Analysis

The region-based multi-resolution analysis can be achieved using downsampling expansion with maximum energy extraction followed by upsampling reconstruction.

Down sampling expansion with maximum energy extraction: the highest resolution image is reconstructed from the root of the coefficients tree, each group of lower resolution image is reconstructed respectively through each group of relative lower resolution coefficients sets. Since the maximum energy is contained within the lowest frequency coefficients; we do maximum energy extraction by keeping the lowest P coefficients, that is computing image only based on the lowest P coefficients.

Up sampling reconstruction: each group of lower resolution reconstructed images are merged to form a higher resolution parent reconstructed image and this procedure is repeated until we arrive at the root.

We still take Cartesian Zernike moments as an example. This region-based multi-resolution analysis procedure is shown in Figure 3.5. Cartesian Zernike moments do not transfer an image into the frequency domain. The essential information of an image is not only contained by the coefficients with the lower order. Therefore, in this example, we skip the maximum energy extraction. However, we can do maximum energy extraction based on selecting important coefficients (features); in chapter four, we will introduce a novel algorithm to select these important coefficients (features).

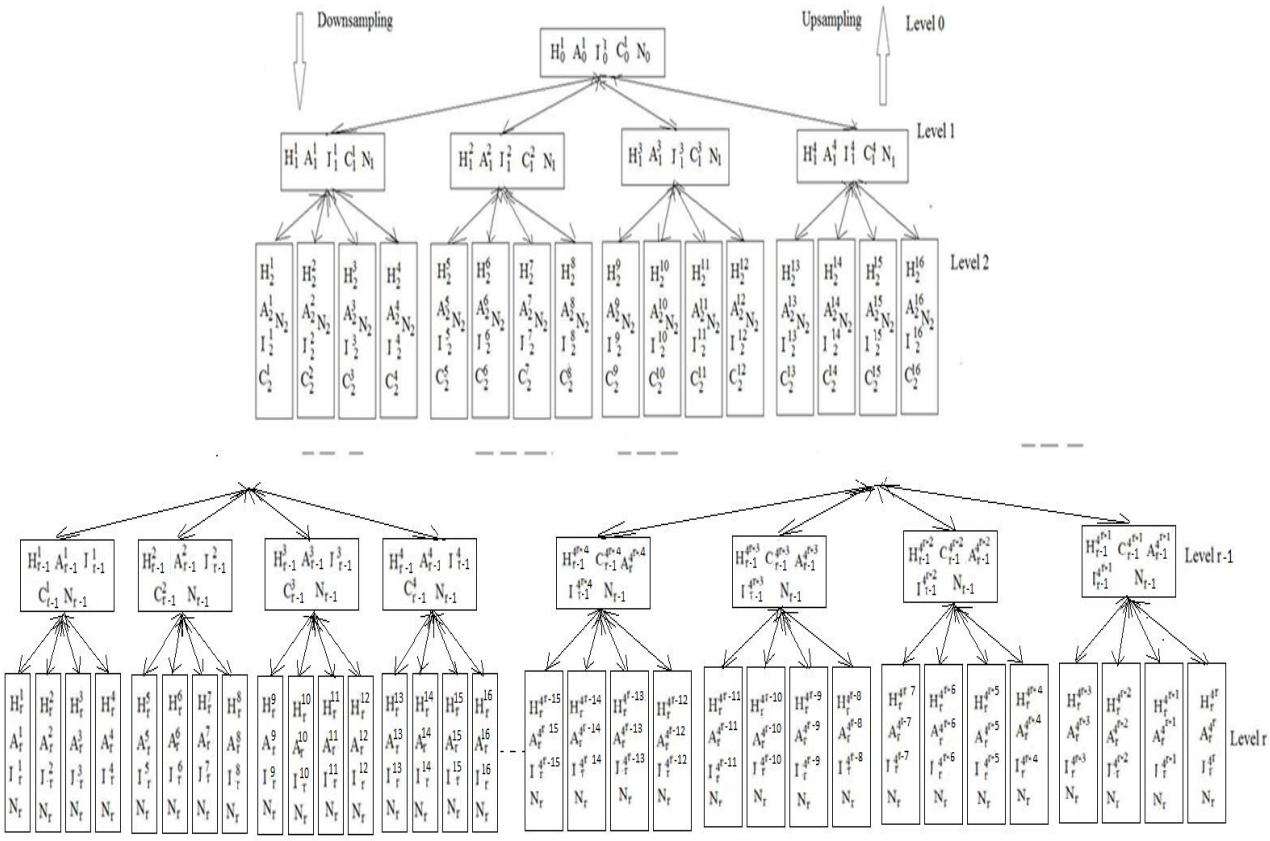

Figure 3.5 Region-based Multi-resolution Analysis

H: image from odd-even image region tree; A: multi-resolution Cartesian Zernike moments;

I: downsampling expanded image ; C: upsampling reconstructed image;

N: number of pixels in an image

(1) Region-based downsamping expantion

The downsampling expanded image I_r^m is computed as follows:

$$I_r^m((k_1)_r^m,(k_2)_r^m) = \sum_{n=0}^{Q}\sum_{L=0}^{n}(A_{nL})_r^m\, V_{nL}((k_1)_r^m,(k_2)_r^m) \qquad (3.21)$$

Where n : order, non-negative integer;

$$L: non\ negative\ integer\ n-L\ is\ even\ and\ L\ \leq\ n$$

$$(k_1)_r^m = 0,1,2,\dots(K_1)_r - 1; (K_1)_r = \frac{K_1}{2^r};\ (k_2)_r^m = 0,1,2,\dots(K_2)_r - 1;\ (K_2)_r = \frac{K_2}{2^r};$$

$$m = 1, 2, 3, \dots, 4^r; r = 0, 1, 2, \dots .$$

When r $= 0$, we get the root of the downsampling expanded image:

$$I_0^1((k_1)_0^1, (k_2)_0^1) = \sum_{n=0}^{Q} \sum_{L=0}^{n} (A_{nL})_0^1 \, V_{nL}((k_1)_0^1, (k_2)_0^1) \tag{3.22}$$

$$Where\ (k_1)_0^1 = 0, 1, 2, \dots, K_1 - 1\ ;\ (k_2)_0^1 = 0, 1, 2, \dots, K_2 - 1.$$

When r > 0, we get 4^{r-1} groups of odd-odd, odd-even, even-odd and even-even downsampling expanded image $I_r^{m_{oo}}$, $I_r^{m_{oe}}$, $I_r^{m_{eo}}$ and $I_r^{m_{ee}}$:

$$\begin{cases} I_r^{m_{oo}}\left((k_1)_r^{m_{oo}}, (k_2)_r^{m_{oo}}\right) = \sum_{n=0}^{Q} \sum_{L=0}^{n} (A_{nL})_r^{m_{oo}} \, V_{nL}\left((k_1)_r^{m_{oo}}, (k_2)_r^{m_{oo}}\right) \\ I_r^{m_{oe}}\left((k_1)_r^{m_{oe}}, (k_2)_r^{m_{oe}}\right) = \sum_{n=0}^{Q} \sum_{L=0}^{n} (A_{nL})_r^{m_{oe}} \, V_{nL}\left((k_1)_r^{m_{oe}}, (k_2)_r^{m_{oe}}\right) \\ I_r^{m_{eo}}\left((k_1)_r^{m_{eo}}, (k_2)_r^{m_{eo}}\right) = \sum_{n=0}^{Q} \sum_{L=0}^{n} (A_{nL})_r^{m_{eo}} \, V_{nL}\left((k_1)_r^{m_{eo}}, (k_2)_r^{m_{eo}}\right) \\ I_r^{m_{ee}}\left((k_1)_r^{m_{ee}}, (k_2)_r^{m_{ee}}\right) = \sum_{n=0}^{Q} \sum_{L=0}^{n} (A_{nL})_r^{m_{ee}} \, V_{nL}\left((k_1)_r^{m_{ee}}, (k_2)_r^{m_{ee}}\right) \end{cases} \tag{3.23}$$

$$\text{Where} (k_1)_r^{m_{oo}} = (k_1)_r^{m_{oe}} = (k_1)_r^{m_{eo}} = (k_1)_r^{m_{ee}} = 0, 1, 2, 3, \dots, (K_1)_r - 1;$$

$$(k_2)_r^{m_{oo}} = (k_2)_r^{m_{oe}} = (k_2)_r^{m_{eo}} = (k_2)_r^{m_{ee}} = 0, 1, 2, 3, \dots, (K_2)_r - 1;$$

$$m_{oo} = 1,5, 9, 13, 17, \dots, 4^r - 3; m_{oe} = m_{oo} + 1;\ m_{eo} = m_{oo} + 2; m_{ee} = m_{oo} + 3.$$

(2)Region-based upsampling reconstruction

When $r = R - 1$ where R is the height of the odd-even image region tree, we get 4^{R-1} upsampling reconstructed image by merging 4^{R-1}groups of odd-odd, odd-even, even-odd and even-even downsampling expanded image $I_R^{m_{oo}}$, $I_R^{m_{oe}}$, $I_R^{m_{eo}}$ and $I_R^{m_{ee}}$ from lower resolution level:

$$C_{R-1}^{h}\left((k_1)_{\mathrm{R}-1}^{\mathrm{h}},(k_2)_{\mathrm{R}-1}^{\mathrm{h}}\right)$$

$$=\begin{cases} I_R^{\mathrm{moo}}\left((k_1)_R^{\mathrm{moo}},(k_2)_R^{\mathrm{moo}}\right) & (k_1)_{\mathrm{R}-1}^{\mathrm{h}}=2i_1,(k_2)_{\mathrm{R}-1}^{\mathrm{h}}=2i_2 \\ I_R^{\mathrm{moe}}\left((k_1)_R^{\mathrm{moe}},(k_2)_R^{\mathrm{moe}}\right) & (k_1)_{\mathrm{R}-1}^{\mathrm{h}}=2i_1,(k_2)_{\mathrm{R}-1}^{\mathrm{h}}=2i_2+1 \\ I_R^{\mathrm{meo}}\left((k_1)_R^{\mathrm{meo}},(k_2)_R^{\mathrm{meo}}\right) & (k_1)_{\mathrm{R}-1}^{\mathrm{h}}=2i_1+1,(k_2)_{\mathrm{R}-1}^{\mathrm{h}}=2i_2 \\ I_R^{\mathrm{mee}}\left((k_1)_R^{\mathrm{mee}},(k_2)_R^{\mathrm{mee}}\right) & (k_1)_{\mathrm{R}-1}^{\mathrm{h}}=2i_1+1,(k_2)_{\mathrm{R}-1}^{\mathrm{h}}=2i_2+1 \end{cases} \quad (3.24)$$

Where $(k_1)_R^{\mathrm{moo}}=(k_1)_R^{\mathrm{moe}}=(k_1)_R^{\mathrm{meo}}=(k_1)_R^{\mathrm{mee}}=i_1=0,1,2,3,\dots,(K_1)_{\mathrm{R}}-1;\ (K_1)_R=\frac{K_1}{2^R}$;

$$(k_2)_R^{m_{oo}}=(k_2)_R^{m_{oe}}=(k_2)_R^{m_{eo}}=(k_2)_R^{m_{ee}}=i_2=0,1,2,3,\dots,(K_2)_R-1;\ (K_2)_R=\frac{K_2}{2^R};$$

$$(k_1)_{R-1}^{h}=0,1,2,3,\dots,(K_1)_{R-1}-1;(K_1)_{R-1}=\frac{K_1}{2^{R-1}};$$

$$(k_2)_{R-1}^{h}=0,1,2,3,\dots,(K_2)_{R-1}-1;(K_2)_{R-1}=\frac{K_2}{2^{R-1}}$$

$$m_{oo}=1,5,9,13,17,\dots,4^R-3;m_{oe}=m_{oo}+1;\ m_{eo}=m_{oo}+2;m_{ee}=m_{oo}+3;$$

$$h=1,2,3,\dots,4^{R-1}.$$

We consider these 4^{R-1} upsampled images as 4^{R-2} groups of odd-odd, odd-even, even-odd, even-even upsampling reconstructed images $C_{R-1}^{h_{oo}}$, $C_{R-1}^{h_{oe}}$, $C_{R-1}^{h_{e0}}$and $C_{R-1}^{h_{ee}}$ where $h_{oo}=1,5,9,13,17,\dots,4^{R-1}-3;h_{oe}=h_{oo}+1;\ h_{eo}=h_{oo}+2;h_{ee}=h_{oo}+3.$

When $r<R-1$, we get 4^{r-1} groups of odd-odd, odd-even, even-odd, even -even upsampling reconstructed images $C_r^{h_{oo}}$, $C_r^{h_{oe}}$, $C_r^{h_{e0}}$and $C_r^{h_{ee}}$ where $h_{oo}=1,5,9,13,17,\dots,4^r-3;h_{oe}=h_{oo}+1;\ h_{eo}=h_{oo}+2;\ h_{ee}=h_{oo}+3$ by merging 4^rgroups of odd-odd, odd-even, even-odd, even -even upsampling reconstructed images $C_{r+1}^{h'_{oo}}$, $C_{r+1}^{h'_{oe}}$, $C_{r+1}^{h'_{eo}}$and $C_{r+1}^{h'_{ee}}$ from lower resolution level. The merging rule is the same except that we merge $C_{r+1}^{h'_{oo}}$, $C_{r+1}^{h'_{oe}}$, $C_{r+1}^{h'_{eo}}$and $C_{r+1}^{h'_{ee}}$ instead of $I_R^{m_{oo}}$, $I_R^{m_{oe}}$, $I_R^{m_{eo}}$ and $I_R^{m_{ee}}$. This procedure is repeated until arriving at the root where

we get C_0^1 with the same size of the original image.

We can apply our region-based multi-resolution scheme to do multi-resolution analysis with other region-based multi-resolution descriptors under the same procedure, except replacing multi-resolution Cartesian Zernike moments with other region-based multi-resolution descriptors and adding maximum energy extraction in case. Furthermore the procedures of downsampling and upsampling have tree structures that are ideal for parallel computing.

3.3 Characteristics of Multi-resolution Scheme

The characteristics of our multi-resolution scheme are as follows:

- The multi-resolution concepts provide both spatial and frequency domain access to the analysis of images recursively.
- The lower-resolution images can be used for analysis of large structures or overall image context; the higher-resolution images are appropriate for analyzing individual object characteristics. Such a coarse-to-fine analysis strategy is useful in pattern recognition.
- The down sampling expansion produces different images to satisfy different application requirements. The up sampling reconstruction constructs higher resolution level images using information from lower resolution levels, which is helpful when some image information is missing.
- Maximum energy extraction contains the essential information for images, saves computation resources and improves computation efficiency.
- The selection of odd-even pixels is ideal for compressing data and filtering noise, making our multi-resolution scheme effective and robust for real applications.

- The odd-even image tree can produce both boundary-based and region-based multi-resolution image descriptors, which greatly generalizes the applications of our multi-resolution scheme.
- Our multi-resolution scheme only uses the information from the original image without requiring other theories or prior-knowledge, which improves the accuracy of the image representation and analysis.
- Our multi-resolution scheme is ideal for parallel computing which makes the proposed method more prospective for large image database processing.

Summary

In this chapter, we propose our novel multi-resolution image representation and analysis scheme. Through applying the image descriptors to the odd-even image tree, we can get the relative multi-resolution image descriptors. The multi-resolution analysis is achieved based on downsampling expansion with maximum energy extraction followed by up sampling reconstruction. Our scheme has two formats: a boundary-based scheme that generates boundary-based multi-resolution image descriptors and a region-based scheme that generates the region-based multi-resolution descriptors. Our scheme can analyze images recursively and effectively without introducing artifacts or changes to the original image, produce different kinds of multi-resolution representations, obtain higher resolution images only using information from lower resolution levels, and be implemented in parallel processing.

4 FEATURE EXTRACTION BASED ON MULTI-RESOLUTION SCHEME

In this Chapter, we introduce a novel image feature extraction algorithm based on our multi-resolution scheme. Our algorithm has advanced computation accuracy and efficiency; it can be adapted to different kinds of image descriptors.

4.1 Review of Current Feature Extraction Algorithms

When analyzing complex data, one of the major problems comes from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power; it may make a classification algorithm overfit the training sample and generalize poorly to the new samples. Feature extraction is a general methods for reducing the amount of the variables with describing the data with sufficient accuracy [38].

As to pattern recognition and image processing, feature extraction is a special form of dimension reduction. An image descriptor G usually uses a set of coefficients (features) , that is $G = \{g_1, g_2, \dots, g_w, \dots, g_W\}$ to describe an image. Some of these coefficients are important; some are redundant; to improve the computation accuracy and efficiency, we do feature extraction that is extracting important features and ignoring unimportant ones.

Now we give a brief review of current popular feature extraction algorithms.

The most popular linear method for feature extraction is principal component analysis (PCA) that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables that are considered as principal components [116]. Principal components are guaranteed to be independent if the input data is jointly normally distributed. PCA is sensitive to the relative scaling of the original variables; it also has two pre-requirements: (1) the dimensionality of data can be efficiently

reduced by linear transformation; (2) most information is contained in those directions where input data variance is maximum[116]. These pre-requirements cannot be always achieved.

PCA can be employed in a nonlinear way by kernel trick [117]. The resulting technique is capable of constructing nonlinear mappings to maximize the variance. This technique is kernel PCA [118]. However, kernel PCA still has very high computation complexity.

Other prominent nonlinear techniques include manifold learning techniques [119] such as Isomap [120], locally linear embedding (LLE) [121], Hessian LLE [122], Laplacian eigenmaps [122], and Local tangent space alignment (LTSA) [123]. These techniques construct a low-dimensional data representation based on a cost function; this cost function holds local properties of the data and can be considered as defining a graph-based kernel for Kernel PCA. One limitation of these algorithms is that the design of data dependent kernel matrices for various manifold learning algorithms is a nontrivial task.

Recently, instead of defining a fixed kernel, semidefinite embedding (SDE) or maximum variance unfolding (MVU) uses semidefinite programming to do non-linear reduction for high-dimensional vectorial input data [124]. MVU can be considered as a non-linear generalization of PCA. SDE has better performance in revealing the underlying dimension of the data than LLE and Laplacian eigenmaps; it guarantees that the nearest neighbors in the embedding is as same as the original nearest neighbor for each point while the other two methods do not. However, SDE is much slower and harder to scale to large data [124].

Another popular method for feature selection is autoencoders, that is a special kind of feed-forward neural networks with a bottle-neck hidden layer [125]. The training of deep encoders is usually performed using a greedy layer-wise pre-training followed by a back

propagation finetuning stage. However auto encoders have high computation complexity and requires advanced hardware.

The current feature extraction algorithms are either restricted by assumptions or hard to be implemented or suffering high computation complexity. Furthermore, these methods are not based on multi-resolution image analysis that usually reveals the hidden information of images

4.2 Feature Extraction Algorithm based on Multi-resolution Scheme

Exploring features in different resolutions reveals the dominant features (coefficients) in comparison to the redundant ones. Based on our multi-resolution scheme, we propose a novel feature extraction algorithm.

Algorithm I: Feature Extraction based on Multi-resolution Scheme

Input: an image database: image im_v $(v = 1, 2, \dots, V)$

an image descriptor: $G = \{g_1, g_2, \dots, g_w, \dots, g_W\}$

Output: the extract features (coefficients) $G' = \{\mathrm{g}'_1, \mathrm{g}'_2, \ \dots, \mathrm{g}'_P\}$ $\quad (P \leq W)$

Step one: transform each image im_v into an odd-even image tree with height R.

Step two: apply the image descriptor $G = \{g_1, g_2, \dots, g_w, \dots, g_W\}$ to the odd-even image tree relative to image im_v to get the relative multi-resolution image descriptor $(\mathrm{G}_r^m)_v = \{((g_1)_r^m)_v, ((g_2)_r^m)_v, ((g_3)_r^m)_v, \dots, ((g_w)_r^m)_v, \dots, ((g_W)_r^m)_v\}$

Step three: for each coefficient g_w of G, compute its two variances: (i) image variance: variance among different images within the same resolution; (ii) resolution variance: variance among different resolutions within the same image .

Step four: for each coefficient g_w of G, compute its variance ratio $var\_g_w = (image\ variance\ of\ g_w)/(resolution\ \ variance\ of\ g_w)$.

Step five: rank coefficient g_w under the descending order of its variance ratio var_g_w and output the top P coefficients as the effective features that is $G' = \{g'_1, g'_2, \ldots, g'_P\}$.

Our algorithm can be applied to both boundary-based and region-based image descriptors. If G is boundary-based, we transform image im_v into an odd-even image boundary tree; if G is region-based, we transform image im_v into an odd-even image region tree.

Here are the characteristics of our feature extraction algorithm:

- Free of assumptions: our algorithm is adaptive to any kind of image database, without considering the distributions and characters of the input data.
- Parallel computing: the odd-even image tree and coefficients tree are both ideal for parallel processing that can greatly improve the computation efficiency.
- Robust to noise: our algorithm tends to extract the features (coefficients) with lower resolution variance; this preference improves the robust to noise.
- Generalization: since our multi-resolution scheme can be applied into different kinds of image descriptors; our feature extraction algorithm based on this multi-resolution scheme can be also applied into different kinds of image descriptor.

Now we take Cartesian Zernike moments as an example.

In Cartesian Zernike moments, repetition L is decided by order n $(0 \le L \le n, n - L\ is\ even)$, given n a maximum value Q, we have a set of Cartesian Zernike moments with fixed number of coefficients, that is $A_{nL} = \{A_{00}, A_{20}, A_{22}, A_{31}, A_{33}, \ldots, A_{Q\,(Q-2)}, A_{QQ}\}$ where $n = 0, 1, \ldots, Q$. Since it is difficult to define the most effective Zernike coefficients in a concise way;

how to extract an effective subset of Zernike coefficients from all possible coefficients under Zernike order n is an open problem.

The most popular method is using all coefficients as image features. This method increases computation complexity and may generate computational errors due to the redundant features.

An alternate method is to use the first P coefficients. However, Cartesian Zernike moments do not transfer the image into the frequency domain. The essential information is not only contained by the coefficients with the lower order. This method arbitrarily ignores the importance of other coefficients; therefore it is not optimized.

Exploring Zernike moments in different resolutions reveals the dominant coefficients (features) in comparison to the redundant ones. Here we use our feature extraction algorithm based on multi-resolution Cartesian Zernike moments to extract essential Cartesian Zernike coefficients [110].

Algorithm II: Feature Extraction based on Multi-resolution Cartesian Zernike Moments

Input: an image database: image im_v with the size $(K_1)_v \times (K_2)_v \quad (v = 1, 2, \dots, V)$

Cartesian Zernike moments $A_{nL} = \{a_1, \dots, a_W\}$ (W is decided by maximum value of n)

Output: extracted Cartesian Zernike coefficients set: $A'_{nL} = \{a'_1, a'_2, \dots, a'_P\}$ $(P \leq W)$

Step one: transform each image im_v to an odd-even image region tree with height R. The multi-resolution image at node m of tree level r is represented by

$(H_r^m)_v(((k_1)_r^m)_v, ((k_2)_r^m)_v)$ where $((k_1)_r^m)_v = 0, 1, 2, \dots ((K_1)_r)_v - 1;\ ((K_1)_r)_v = \frac{(K_1)_v}{2^r};$

$((k_2)_r^m)_v = 0, 1, 2, \dots ((K_2)_r)_v - 1;\ ((K_2)_r)_v = \frac{(K_2)_v}{2^r}; m = 1, 2, 3, \dots, 4^r;\ r = 0,1,2, \dots, R.$

Step two: for each image im_v, generate its multi-resolution Cartesian Zernike moments by applying traditional Cartesian Zernike moments to its odd-even image region tree.

$$((A_{nL})_r^m)_v = \frac{n+1}{\pi|(H_r^m)_v(((k_1)_r^m)_v,((k_2)_r^m)_v)|}\sum_{((k_1)_r^m)_v=0}^{((K_1)_r)_v-1}\sum_{((k_2)_r^m)_v=0}^{((K_2)_r)_v-1}(H_r^m)_v(((k_1)_r^m)_v,((k_2)_r^m)_v)$$

$$\times V_{nL}^*(((k_1)_r^m)_v,((k_2)_r^m)_v)$$

The symbol * denotes the complex conjugate of V_{nL} .

Where $V_{nL}(((k_1)_r^m)_v,((k_2)_r^m)_v)$ *is Cartesian Zernike moments polynomials for* im_v

$$V_{nL}(((k_1)_r^m)_v,((k_2)_r^m)_v)$$

$$= R_{nL}(((k_1)_r^m)_v,((k_2)_r^m)_v)\Bigg(((((k_1)_r^m)_v)^2$$

$$+ (((k_2)_r^m)_v)^2)^{\frac{-L}{2}}\sum_{f=0}^{m_d}(-1)^f\binom{L}{2f}(((k_1)_r^m)_v)^{L-2f}(((k_2)_r^m)_v)^{2f}$$

$$+ j((((k_1)_r^m)_v)^2$$

$$+ (((k_2)_r^m)_v)^2)^{\frac{-L}{2}}\sum_{f=0}^{m_q}(-1)^f\binom{L}{2f+1}(((k_1)_r^m)_v)^{L-2f-1}(((k_2)_r^m)_v)^{2f+1}\Bigg)$$

$j=\sqrt{-1}$ *; n: order, non-negative integer;*

L: repetition, non negative integer subject to the constraints $n-L$ *is even and* $L \le n$.

$$R_{nL}(((k_1)_r^m)_v,((k_2)_r^m)_v)$$

$$= \sum_{s=0}^{(n-L)/2}(-1)^s\frac{(n-s)!}{s!\left(\frac{n+L}{2}-s\right)!\left(\frac{n-L}{2}-s\right)!}(((((k_1)_r^m)_v)^2$$

$$+ (((k_2)_r^m)_v)^2)^{\frac{n-2s}{2}}$$

$m_d$and m_q are defined with respect to L:

when L is even $m_d = \frac{L}{2}, m_q = \frac{L-2}{2}$*; when L is odd* $m_d = \frac{L-1}{2}, m_q = \frac{L-1}{2}$

$$|(H_r^m)_v(((k_1)_r^m)_v, ((k_2)_r^m)_v)| = ((K_1)_r)_v \times ((K_2)_r)_v$$

$$m = 1, 2, 3, \dots, 4^r;\ r = 0,1,2, \dots, R.$$

$((A_{nL})_r^m)_v$ can be considered as a coefficients tree; at node m of tree level r there is coefficients set $((A_{nL})_r^m)_v = \{((a_1)_r^m)_v, ((a_2)_r^m)_v, \dots, ((a_w)_r^m)_v, \dots, ((a_W)_r^m)_v\}$; W is decided by the maximum value of order n.

Step three: for each a_w of Cartesian Zernike moments $A_{nL} = \{a_1, \dots, a_w, \dots, a_W\}$, compute its two variances: (i) image variance: variance among different images within the same resolution;(ii) resolution variance: variance among different resolutions within the same image

(i) Image variance.

In the coefficients tree relative to image im_v, a Cartesian Zernike coefficient a_w is expressed as $((a_w)_r^m)_v$ where $r = 0, 1, 2, \dots, R$ and $m = 1, 2, 3, \dots, 4^r$. We compute the image variance of $((a_w)_r^m)_v$ among V different images within the tree level r:

$$(varim\_a_w)_r^m = \frac{1}{V-1}\sum_{v=1}^{V}(((a_w)_r^m)_v - \frac{1}{V}\sum_{v=1}^{V}((a_w)_r^m)_v)^2 \quad (4.1)$$

Where $w = 1, 2, 3, \dots, W$.

We consider the average value of $(varim\_a_w)_r^m$ as the image variance of a_w:

$$avg\_varim\_a_w = \frac{3}{4^{R+1}-1}\sum_{r=0}^{r=R}\sum_{m=1}^{4^r}(varim\_a_w)_r^m \quad (4.2)$$

Where $w = 1, 2, 3, \dots, W$.

(ii) Resolution variance.

In the coefficients tree with height R, the lowest resolutions of a Cartesian Zernike coefficient a_w are $((a_w)_R^m)_v$ where $m = 1, 2, 3, \dots, 4^R$. We compute the image variance of $((a_w)_r^m)_v$ among $R + 1$ resolution levels within the same image im_v:

$$((varre\_a_w)_R^m)_v = \frac{1}{R}\sum_{r=0}^{R}\left(((a_w)_r^m)_v - \frac{1}{R+1}\sum_{r=0}^{R}((a_w)_r^m)_v\right)^2 \tag{4.3}$$

Where $w = 1, 2, 3, \dots, W$.

We consider the average value of $((varre\_a_w)_R^m)_v$ as the resolution variance of a_w:

$$avg\_varre\_a_w = \frac{1}{V \times 4^R}\sum_{v=1}^{v=V}\sum_{m=1}^{4^R}((varre\_a_w)_R^m)_v \tag{4.4}$$

Where $w = 1, 2, 3, \dots, W$.

The high image resolution of a coefficient shows its outstanding ability to distinguish different images. The low resolution variance of a coefficient demonstrates its strong ability to represent the main characteristics of individual image.

Step four: for each Cartesian Zernike coefficient a_w , compute its variance ratio

$$var\_a_w = avg\_varim\_a_w / avg\_varre\_a_w.$$

Step five: rank the Cartesian Zernike coefficient a_w under the descending order of var_a_w and output the top P coefficients as the effective features that is $A'_{nL} = \{a'_1, a'_2, \dots, a'_P\}$.

Now we take simulations to demonstrate the effectives of our feature extraction algorithm; we still use Cartesian Zernike moments as an example.

(1) Simulation data

We use three true biomedical data sets for simulation [115]:

(i) Head-Neck Cetuximab: its subject ID is 0522c0002 and its series ID is 1.3.6.1.4.1.14519.5.2.1.5099.8010.217836670708542506360829799868. This image database has 404 images in total. The samples of this image set are shown in Figure 4.1.

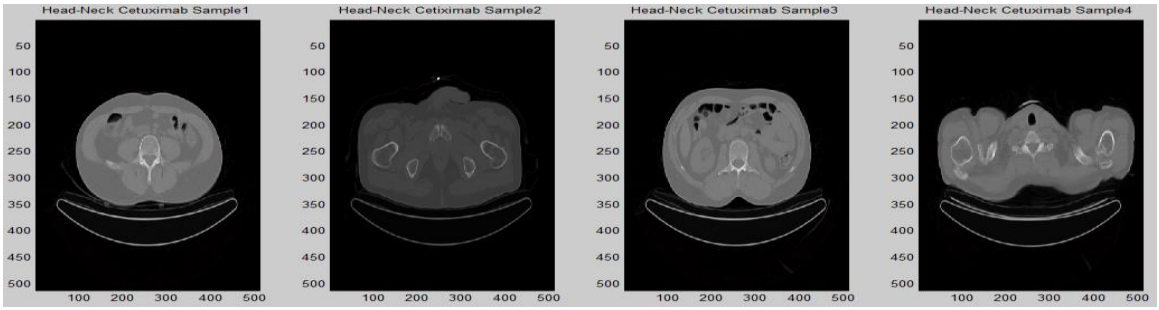

Figure 4.1 Samples of 'Head-Cetuximab'

(ii)BREAST-DIAGNOSIS: its object ID is BreastDx-01-0002 and its series ID is 1.3.6.1.4.1.14519.5.2.1.4792.2001.274817214414807922285870207347. This image database has 207 images in total. The samples of this image set are shown in Figure 4.2.

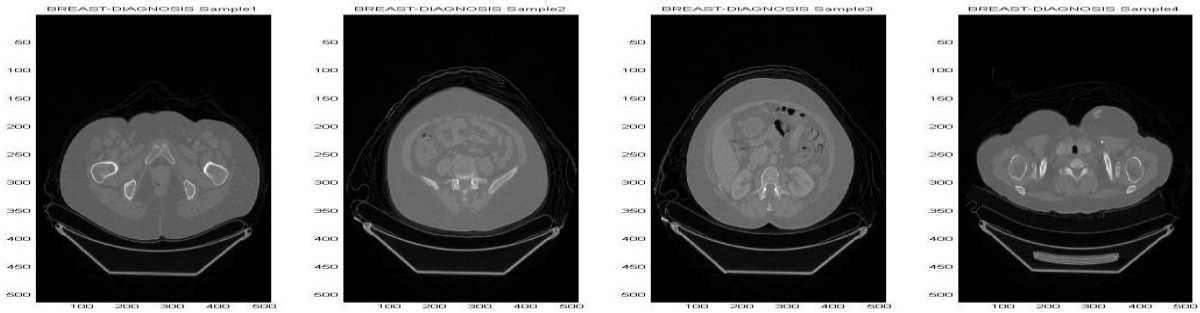

Figure 4.2 Samples of ‘BREAST-DIAGNOSIS’

(iii) REMBRANDT: its object ID is 900-00-5308 and its series ID is 1.3.6.1.4.1.9328.50.50.332219323434404597214250726745797089409. This image database has 80 images in total. The samples of this image set are shown in Figure 4.3.

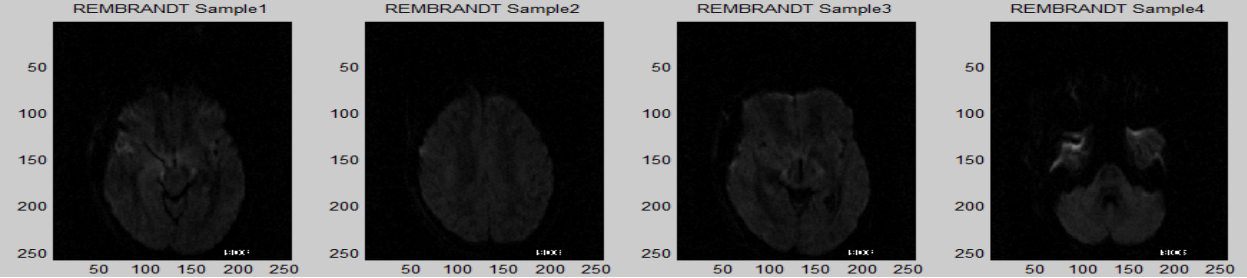

Figure 4.3 Samples of 'REMBRANDT'

For each data set, we randomly choose half set for training and leave anther half for test. We also add Gaussian, "Salt & Pepper" and speckle noises to each test set respectively.

(2) Simulation Process

We set Zernike order as 10; therefore we have 36 Zernike coefficients in total.

We use our algorithm to respectively analyze each of the three training sets in multi-resolution levels ($r = 0, 1, 2$) and extract 18 effective coefficients out of the total 36 coefficients.

Based on 'Head-Neck Cetuximab' training set, we extract 18 coefficients $\{A_{0\,0}, A_{2\,0}, A_{2\,2}, A_{4\,2}, A_{4\,4}, A_{7\,7}, A_{4\,0}, A_{8\,8}, A_{6\,6}, A_{5\,5}, A_{6\,4}, A_{8\,2}, A_{10\,10}, A_{7\,5}, A_{3\,3}, A_{8\,6}, A_{6\,0}, A_{6\,2}\}$. We apply each extracted coefficient to 'Head-Neck Cetuximab' test set without noise, 'Head-Neck Cetuximab' test set with Gaussian noise, 'Head-Neck Cetuximab' test set with 'Salt & Pepper' noise, and 'Head-Neck Cetuximab' test set with Speckle noise to compute its variance among images within each test set respectively. These image variances are shown in Table 4.1. In the last row of Table 4.1, we compute mean of image variances generated by these extracted 18 coefficients and consider this mean as the image variance of the extracted coefficients set.

Based on BREAST-DIAGNOSIS training set, we extract 18 coefficients $\{A_{00}, A_{42}, A_{60}, A_{64}, A_{62}, A_{22}, A_{40}, A_{10\,2}, A_{80}, A_{82}, A_{20}, A_{33}, A_{55}, A_{86}, A_{31}, A_{53}, A_{10\,4}, A_{84}\}$. We apply each extracted coefficient to 'BREAST-DIAGNOSIS' test set without noise, 'BREAST-DIAGNOSIS' test set with Gaussian noise, 'BREAST-DIAGNOSIS' test set with 'Salt &

Pepper' noise, and BREAST-DIAGNOSIS test set with Speckle noise to compute its variance among different images within each test set respectively. These image variances are shown in Table 4.2. In the last row of Table 4.2, we compute the mean of image variances of the extracted 18 coefficients and consider this mean as image variance of the extracted coefficients set.

Based on 'REMBRANDT' training set, we extract 18 coefficients $\{A_{0\,0}, A_{2\,0}, A_{3\,1}, A_{4\,2}, A_{1\,1}, A_{2\,2}, A_{10\,6}, A_{8\,4}, A_{10\,4}, A_{4\,0}, A_{6\,2}, A_{5\,3}, A_{8\,6}, A_{6\,4}, A_{9\,1}, A_{5\,5}, A_{10\,2}, A_{9\,5}\}$ We apply each extracted coefficient to 'REMBRANDT' test set without noise, 'REMBRANDT' test set with Gaussian noise, 'REMBRANDT' test set with 'Salt & Pepper' noise, and REMBRANDT' test set with Speckle noise, to compute its variance among different images within each test set respectively. These image variances are shown in Table 4.3. In the last row of Table 4.3, we compute the mean of image variances of these extracted 18 coefficients and consider this mean as the image variance of the extracted coefficients set.

To make comparison, we also compute the image variance of the first 18 coefficients set and the image variance of all 36 coefficients set, for the three kinds of image test sets, 'Head-Neck Cetuximab', 'BREAST-DIAGNOSIS', 'REMBRANDT', respectively. These image variances with the image variances of our extracted coefficients set are shown in Table 4.4.

Higher image variance of a coefficients set demonstrates its stronger ability to reveal the essential characteristics of an image and distinguish this image with other images. Through comparing the image variances of our extracted coefficients set, the first 18 coefficients set, and all 36 coefficients set, we can find the most effective coefficients set.

(3) Simulation Results

Simulation results are shown in Table 4.1-4.4 and Figure 4.4-4.6.

Table 4-1 Image Variance based on Our Extracted Coefficients for 'Head-Neck Cetuximab'

Coefficient	Clean	Gaussian	Salt &Pepper	Speckle
$A_{0\,0}$	246.64994	76.324968	247.91207	246.48968
$A_{2\,0}$	1586.6531	447.11513	1622.876	1586.5919
$A_{2\,2}$	130.68501	42.147868	123.57683	130.70401
$A_{4\,2}$	389.72806	111.14731	370.79883	390.09908
$A_{4\,4}$	117.13405	38.756105	106.60947	117.25542
$A_{7\,7}$	89.074607	20.145987	77.934048	89.116653
$A_{4\,0}$	754.07145	217.25	763.30788	754.79251
$A_{8\,8}$	88.59183	21.26171	68.867576	88.683763
$A_{6\,6}$	111.5449	28.041252	87.142468	111.65107
$A_{5\,5}$	104.36485	26.970637	83.416902	104.29491
$A_{6\,4}$	102.96501	34.681949	104.42016	102.79983
$A_{8\,2}$	377.61894	102.08835	359.23376	378.03856
$A_{10\,10}$	42.877779	15.247046	35.3338	42.901444
$A_{7\,5}$	135.8418	35.092414	123.58442	135.90373
$A_{3\,3}$	119.46275	28.566684	115.06433	119.56129
$A_{8\,6}$	115.90179	27.797151	111.51163	116.11347
$A_{6\,0}$	315.03676	102.02016	306.89698	315.53527
$A_{6\,2}$	205.14689	61.058522	218.79694	205.52215
Average value of Image variances of coefficients	279.6305	79.7618	273.7380	279.7808

Table 4-2 Image Variance based on Our Extracted Coefficients for 'BREAST-DIAGNOSIS'

Coefficient	Clean	Gaussian	Salt &Pepper	Speckle
A_{00}	42.845425	178.4325	42.276361	55.944813
A_{42}	292.4349	1130.2932	292.94925	657.09338
A_{60}	288.74708	1089.1888	291.85701	621.81101
A_{64}	145.73578	507.29323	144.1525	329.66714
A_{62}	182.11926	741.96785	183.74482	423.13098
A_{22}	88.58108	356.85568	91.65745	193.39975
A_{40}	350.60658	1392.3785	357.82252	742.01321
$A_{10\,2}$	370.87685	1226.2637	363.76037	783.81318
A_{80}	357.34973	1403.8164	361.71668	756.08104
A_{82}	169.29564	519.34222	171.64392	358.41842
A_{20}	279.44874	1039.4067	278.50864	517.43558
A_{33}	53.268094	220.741	53.186864	111.23422
A_{55}	18.244497	130.76849	16.002372	38.106757
A_{86}	37.055994	213.25738	39.901875	85.428035
A_{31}	186.47139	743.12979	188.50412	391.82552
A_{53}	64.222353	198.20505	62.421789	136.72578
$A_{10\,4}$	55.355872	338.75276	53.819241	121.94894
A_{84}	81.169741	324.26467	78.144205	188.08109
Average value of Image variances of coefficients	170.2127	653.0199	170.6706	361.7866

Table 4-3 Image Variance based on Our Extracted Coefficients for 'REMBRANDT'

Coefficient	Clean	Gaussian	Salt &Pepper	Speckle
$A_{0\,0}$	350.33569	102.91937	404.52715	350.52338
$A_{2\,0}$	543.01686	199.88941	463.59814	542.65413
$A_{3\,1}$	67.736327	34.599281	82.778873	67.851406
$A_{4\,2}$	15.883407	22.76504	40.016325	15.624189
$A_{1\,1}$	69.248123	29.524644	91.358926	69.396923
$A_{2\,2}$	21.272677	16.88801	36.94541	21.31106
$A_{10\,6}$	14.117759	40.088038	72.794265	13.902909
$A_{8\,4}$	12.779265	38.481738	47.800648	12.769392
$A_{10\,4}$	15.142611	62.029104	46.606309	15.368567
$A_{4\,0}$	48.382452	32.245645	92.33618	48.191253
$A_{6\,2}$	18.744579	22.052078	63.198568	18.619407
$A_{5\,3}$	5.8681866	24.895967	33.119163	5.811925
$A_{8\,6}$	4.2097179	26.140786	37.041634	4.1421563
$A_{6\,4}$	6.4099939	25.292929	42.956971	6.3611321
$A_{9\,1}$	38.258848	32.828728	78.217967	38.542164
$A_{5\,5}$	1.8860031	30.532471	28.721482	1.8941652
$A_{10\,2}$	48.222891	43.037847	86.1374	48.253403
$A_{9\,5}$	8.9820514	45.26694	33.845855	8.7767147
Average value of Image variances of coefficients	71.6943	46.0821	99.0001	71.6663

Table 4-4 Image Variance based on Coefficients Sets

Images database	Algorithm	Image Variance			
		Clean	Gaussian	Salt & Pepper	Speckle
Head-Neck Cetuximab	our extracted coefficients set	**279.6305**	**79.7618**	**273.7380**	**279.7808**
	the first 18 coefficients set	268.2048	78.1616	265.5842	268.3350
	all 36 coefficients set	186.2860	57.5296	185.1490	186.3771
BREAST-DIAGNOSIS	our extracted coefficients set	**170.2127**	**653.0199**	**170.6706**	**361.7866**
	the first 18 coefficients set	119.8462	488.6742	120.7199	253.7421
	all 36 coefficients set	97.2951	418.7415	97.9834	206.9610
REMBRANDT	our extracted coefficients set	**71.6943**	**46.0821**	**99.0001**	**71.6663**
	the first 18 coefficients set	69.3751	41.7063	92.1146	69.2757
	all 36 coefficients set	42.8275	43.2291	77.4969	42.7752

Table 4.4 gives comparisons about the image variance of our extracted coefficients set, the first 18 coefficients set, and all 36 coefficients set based on the three kinds of data sets: 'Head-Neck Cetuximab', 'BREAST-DIAGNOSIS', and 'REMBRANDT'. For all the three kinds of image datasets, whether they are clean or corrupted by noise, our extracted coefficients set always gets the highest image variances; this demonstrates that our algorithm extracts the most effective coefficients. The first 18 coefficients set usually generates higher image variance than the set with all 36 coefficients, except when it processes the 'REMBRANDT' with Gaussian noise; this demonstrates that there are redundant coefficients in the 36 coefficients.

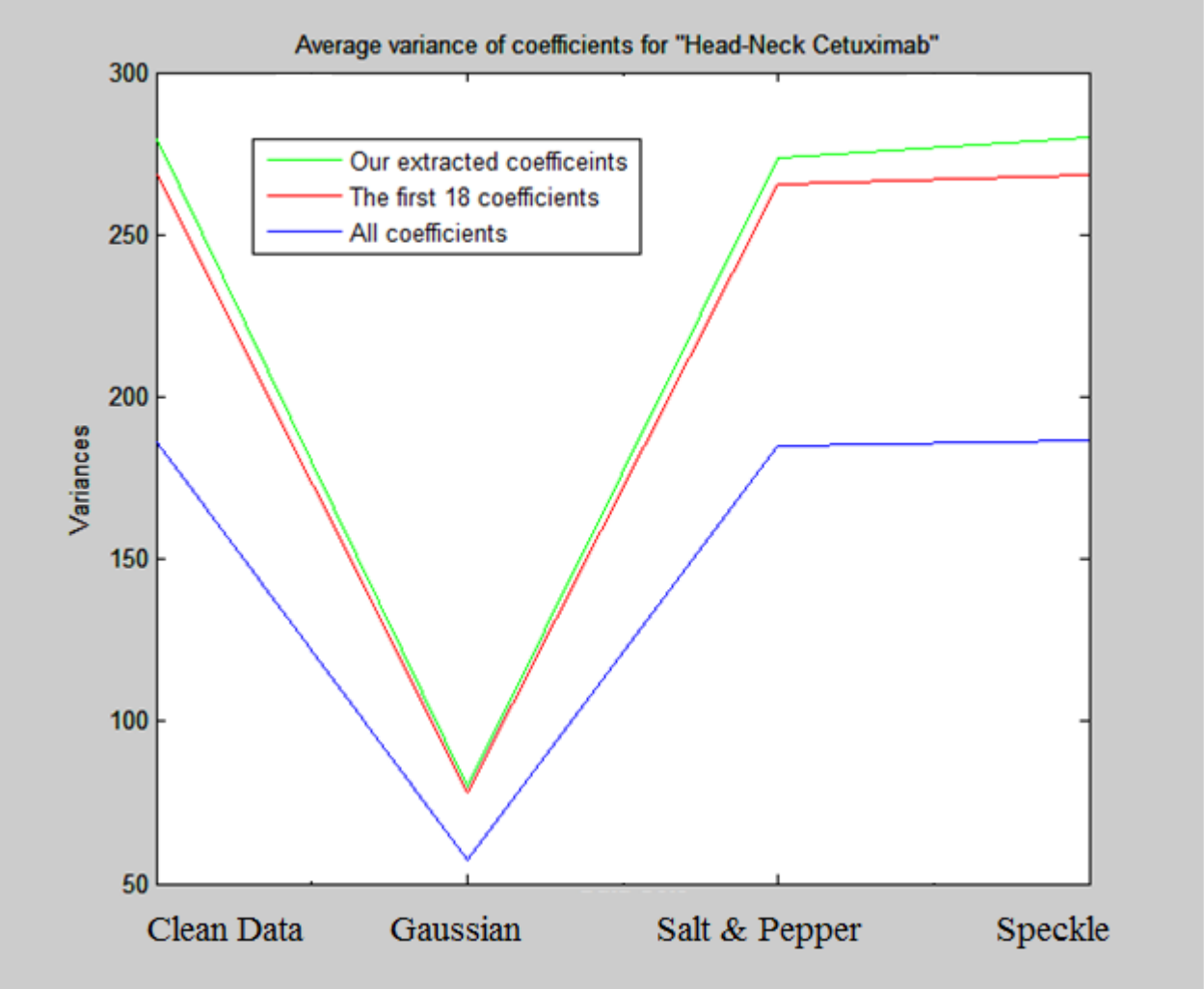

Figure 4.4 Image Variance based on Coefficients Sets for 'Head-Neck Cetuximab'

Figure 4.4 shows image variance based on our extracted coefficients set, the first 18 coefficients set, and all 36 coefficients set for 'Head-Neck Cetuximab'. The green line, representing image variance of our extracted coefficients set, is at the top of the figure. Obviously, our algorithm generates the highest image variance, that is our algorithm has the most effective coefficients to describe the image data set 'Head-Neck Cetuximab'.

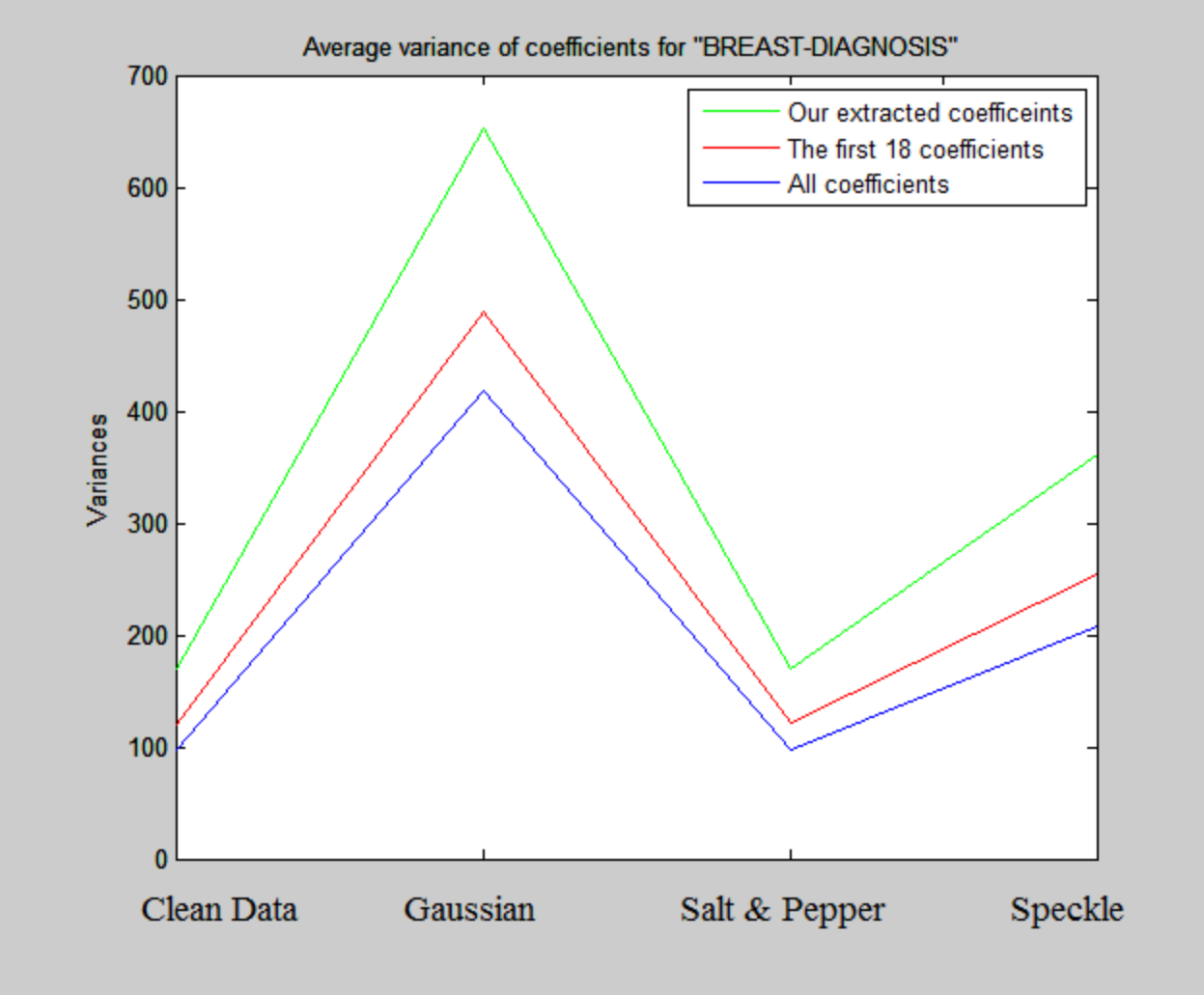

Figure 4.5 Image Variance based on Coefficients Sets for 'BREAST-DIAGNOSIS'

Figure 4.5 shows image variance based on our extracted coefficients set, the first 18 coefficients set, and all 36 coefficients set for 'BREAST-DIAGNOSIS'. The green line, representing image variance of our extracted coefficients set, is at the top of the figure. Obviously, our algorithm generates the highest image variance, that is our algorithm has the most effective coefficients to describe the image data set 'BREAST-DIAGNOSIS'.

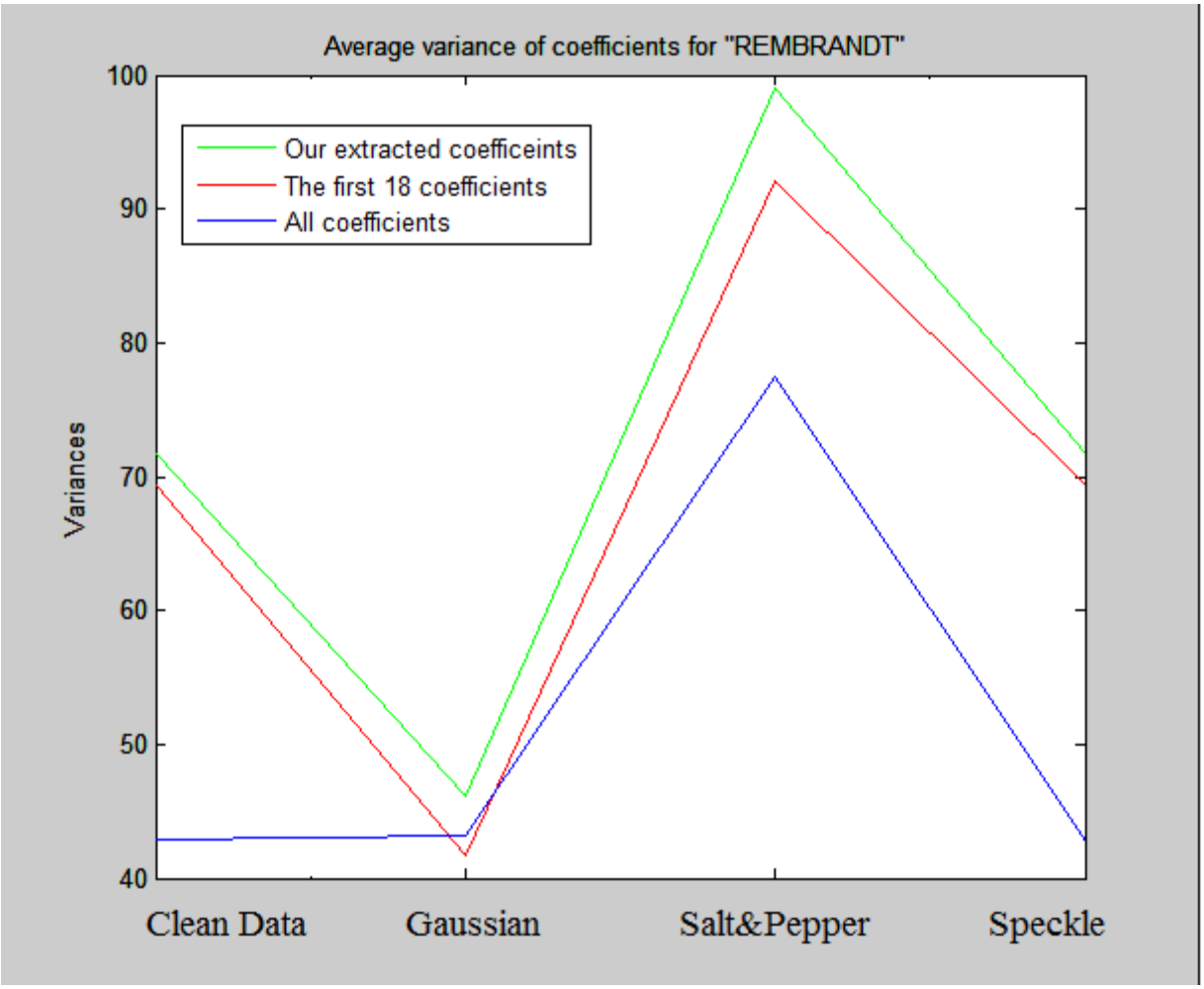

Figure 4.6 Image Variance based on Coefficients Sets for 'REMBRANDT'

Figure 4.6 shows image variance based on our extracted coefficients set, the first 18 coefficients set, and all 36 coefficients set for 'REMBRANDT' The green line, representing image variance of our extracted coefficients set is at the top of the figure. Obviously, our algorithm generates the highest image variance, that is our algorithm has the most effective coefficients to describe the image data set 'REMBRANDT'.

The simulation results show that our extracted coefficients produce the highest image variances because these coefficients represent the dominant characteristics of the image that distinguish this image from others; the set of all 36 coefficients produce the lowest image

variances because the redundant coefficients reduce the overall performance; the image variances produced by the set of the first 18 coefficients are in the middle level because this methods ignores the important coefficients.

The simulation results clearly demonstrate our algorithm extracts excellent features that has superior ability in terms of image representation and identification.

Summary

In this Chapter, we propose a novel image feature algorithm based on our multi-resolution scheme. Our algorithm has advanced computation accuracy and efficiency; it can be adapted into different kinds of image descriptors. We take Cartesian Zernike moments as an example. Our algorithm outperforms the traditional Zernike feature extraction method.

5 IMAGE COMPRESSION AND DENOISING BASED ON MULTI-RESOLUTION SCHEME

In this Chapter, we introduce a novel image compression and denoising algorithm based on our multi-resolution scheme. This algorithm has high computation accuracy and efficiency; it can be adaptive to different kinds of image descriptors.

5.1 Review of Current Image Compression and Denoising Algorithms

5.1.1 Review of Current Image Compression Algorithms

Due to the widespread availability of digital cameras and the rise of the Internet, digital images have become an important method of conveying visual information. The growth of image databases requires considerable storage capacity and transmission bandwidth. Although there is a rapid progress in mass storage density, speed of the processor as well as performance of digital communication systems, the demand for the data storage capacity and data transmission bandwidth continues to exceed the capabilities of the on hand technologies. Moreover, the latest growth of data intensive multimedia based web applications, such as Internet teleconferencing, high definition television, satellite communications and digital storage of movies, etc, are not feasible without a high degree of compression; they are far from realizing their full potential, largely due to the limitations of the traditional image compression methods [126-127, 157].

The goal of image compression is to reduce irrelevance and redundancy of the image data in order to store or transmit data in an efficient form while preserving the quality required for the given application [128-131].

A number of compression schemes based on lossless or lossy criteria have been proposed [128-137]. Although lossless compression [138] is completely reversible, i.e. original data can be

reconstructed from the compressed data without losing information, it can only yield low compression ratios. Therefore, to achieve higher compression ratios, lossy compression should be used. Although lossy compression is irreversible; it eliminates repeated or "unnecessary" pieces of data and maintains visually lossless data that is producing imperceptible differences for people [139].

Various coding schemes have been proposed for lossy data compression, including predictive coding [140], subband coding [141-142], transform coding [143-144], and vector quantization [145-147]. The most popular one is the transform coding. Moreover, discrete cosine transform (DCT) [111, 113, 148-149] and wavelet transform [148, 150-153], are the most popular transform coding methods.

Traditional DCT coding algorithm divides the original image into a set of $K\ by\ K$ blocks, applies DCT to each block and quantizes DCT coefficients of each block. Each image block can be reconstructed by applying invert DCT to its quantized DCT coefficients. Through merging these reconstructed image blocks, we get the decompressed image. DCT coefficients of each block only represents the local information of an image. Separately compressing each block breaks correlation between the pixels at the borders of blocks and cause blocking artifacts [114, 155]. Although DCT has outstanding energy compaction property, the performance of the traditional DCT coders generally degrades at low bit-rates mainly due to its underlying block-based strategy [35, 155]. The well-known JPEG uses a lossy form of compression based on the traditional DCT coding. Although JPEG has been widely used, its DCT blocking artifact is an open problem [35].

Wavelet coding system has the similar structure with the traditional DCT coding system, except the omission of the subimage processing stages of the wavelets transform coder [35]. Since wavelet transforms are inherently local (i.e. their basis functions are limited in duration), subdivision of the original image is unnecessary. The removal of the subdivision step eliminates the blocking artifact that characterizes DCT-based approximations at high compression ratios [35]. Compared with DCT coding, wavelet coding is more robust under the transmission and decoding errors, and better matched to the Human Visual System (HVS) characteristics [148, 153, 156]. Moreover, due to wavelet's inherent multi-resolution nature [151], wavelet coding scheme is especially suitable for applications where scalability and tolerable degradation are important. JEPG 2000 uses a lossy form of compression based on wavelet coding as an extension of JEPG [35, 153]. Unfortunately, wavelets transform has higher computational complexity and requires larger memory [148-149, 156], which is not ideal for supporting big data processing, mobile computing technology and wireless communications.

5.1.2 Review of Current Image Denoising Algorithms

In the real world image databases, noise is usually unavoidable: imperfect instruments, problems with the data acquisition process, and interfering natural phenomena may degrade the quality of data; furthermore, noise can be introduced by transmission errors. Therefore, image denoising is very necessary.

Currently there are two basic approaches to image denoising: spatial filtering methods and transform domain filtering methods.

Under the assumption that the noise occupies the higher region of frequency spectrum, spatial filters apply a low pass filtering on image pixels; it reduces noise to some extent with the

cost of blurring images. Recently, a variety of median type filters such as weighted median [158], rank conditioned rank selection [159], and relaxed median [160] have been proposed to overcome this disadvantage. Mean filter and wiener filtering method are the two popular spatial filters [35, 161]. Mean filter has good performance for Gaussian noise in the sense of mean square error. The wiener filtering method requires the knowledge about the spectra of the noise and the original signal; it works well only if the underlying signal is smooth. Generally speaking, spatial filters tend to blur sharp edges, destroy lines and other fine imaged details; they perform poorly in the presence of signal-dependent noise.

Fourier and wavelets based filtering are two well-known transform domain filtering methods. Generally speaking, wavelets have superior performance in image denoising due to their multi-resolution structure.

Therefore people's focus for image denoising technologies has been shifted from the spatial and Fourier transform domain to the wavelets transform domain. However, wavelets have high computation complexity and poor invariance property that greatly limit wavelets' applications.

Image compression is the process to remove the redundant information; if this removed information is noise, we can do image compression and denoising at the same time. Currently, the most popular image descriptor that can do image compression and denoising simultaneously is wavelets; however wavelets suffer high computation complexity and poor invariance property.

5.2 Image Compression and Denoising Algorithm based on Multi-resolution Scheme

In this section, we propose a novel image compression and denoising algorithm based on our multi-resolution scheme. Our algorithm has high computation accuracy and efficiency; it can be adaptive to different kinds of image descriptor.

Algorithm III: Image Compression and Denoising based on Multi-resolution Scheme

Input: image im

Output: compressed and denoised image G_r'

decompressed and denosied image im_r'

Encoding:

Step one: Transform image im into an odd-even image tree.

Step two: Choose the suitable image descriptor G based on the special case; through applying G to the image tree, generate the relative multi-resolution descriptor G_r^m.

Step three: Construct a coefficients set G_r to represent all coefficients sets in level r.

Step four: Construct G_r' by extracting important coefficients from G_r.

Output G_r' as the compressed and denoised image.

Decoding:

Step five: Generate reconstructed image rm_r by applying relative invert transform to G_r'

Step six: Construct im_r' by resizing reconstructed image rm_r to the original image size.

Output im_r' as the decompressed and denoised image.

Obviously, our algorithm has two main parts: (1) Encoding, from step one to step four, outputs compressed and denoised image G_r'. (2) Decoding: from step five to step six, outputs decompressed and denoised image im_r'.

In the encoding part, the multi-resolution image descriptor G_r^m can be considered as a coefficients tree. If G is boundary-based descriptor, this coefficients tree is a binary tree; at level r, there is 2^r sets of coefficients; each set of coefficients independently represents the original image with $\frac{1}{2^r}$ of its original size in global view and all of these 2^r DCT coefficients sets together lossless represent the original image. If G is region-based descriptor, this coefficients tree is a quarter tree: at level r, there are 4^r sets of coefficients; each set of coefficients independently represents the original image with $\frac{1}{4^r}$ of its original size in global view and all of these 4^r sets of coefficients together lossless represent the original image.

Therefore, in level r, we need to construct a coefficients set G_r to represent all of these 2^r or 4^r sets of coefficients. There are different ways to construct G_r, such as computing the mean of these coefficients sets, choosing the coefficients set with the lowest coefficient variance, etc.. Compared with each set of coefficients in level r, G_r contains more comprehensive information of the original image and is more suitable to globally represent the original image with $\frac{1}{2^r}$ or $\frac{1}{4^r}$ of its original size. As image noise is random variation of information in an image; through computing G_r that can represent all coefficients sets in level r, the noise embedded in the original image is also filtered.

To do further compression and denoising, we extract important coefficients from G_r to build G_r'; G_r' is the compressed and denoised image with a compression ratio that is larger than 2^r or 4^r. We consider G_r' as the output of the encoding part.

In the decoding part, we get the reconstructed image rm_r through applying the related invert transform to G_r'. Then we resize rm_r to the original image size. The resize method is usually based on interpolation which is also a good way to filter noise. We consider the resized image im_r' as the output of the decoding part.

The characteristics of our image compression and denoising algorithm are as following:

- Omission of Blocking Artifact: In our algorithm, each set of coefficients independently represents the original image in global view; this global representation eliminates the blocking artifact.
- Parallel Computing: The odd-even image tree and coefficients tree are ideal for parallel computing.
- Generalization: Since our multi-resolution scheme can be applied into different kinds of image descriptor; our image compression and denoising algorithm based on this multi-resolution scheme can also be applied into different kinds of image descriptors.

Now we use Fourier descriptors as an example for boundary-based image descriptors and discrete cosine transform (DCT) as an example for region-based image descriptors.

5.2.1 Image Compression and Denoising Algorithm based on Multi-resolution Fourier

Fourier descriptors, is a promising boundary-based approach; it has higher computation efficiency than wavelets. However, due to lacking multi-resolution property, traditional Fourier descriptors usually cannot perform as well as wavelets for image compression and denoising. We propose a novel image compression and denoising algorithm based on multi-resolution Fourier. Our algorithm outperforms wavelets in terms of computation accuracy and efficiency.

Algorithm IV: Image Compression and Denoising based on Multi-resolution Fourier

Input: image boundary $S(k)$ $(k = 0, 1, 2, \dots, N-1)$

Output: compressed and denoised image F_r'

decompressed and denoisied image S_r'

Encoding:

Step one: transform image boundary $S(k)$ into an odd-even image boundary tree S_r^m.

$$S_r^m(k_r^m) = x(k_r^m) + jy(k_r^m)$$

Where $k_r^m = 0,1,2, \dots, N_r - 1; N_r = \frac{N}{2^r}; m = 1, 2, 3, \dots, 2^r;\ r = 0, 1, 2, \dots, R; j = \sqrt{-1}$

Step two: Compute multi-resolution Fourier descriptors F_r^m through applying traditional Fourier descriptors to this image boundary tree S_r^m.

$$F_r^m(u_r^m) = \sum_{k_r^m=0}^{N_r-1} S_r^m(k_r^m) e^{-\frac{j2\pi u_r^m k_r^m}{N_r}}$$

Where $u_r^m = 0,1,2, \dots, N_r - 1; N_r = \frac{N}{2^r};\ m = 1, 2, 3, \dots, 2^r; r = 0, 1, 2, \dots, R; j = \sqrt{-1}$

Step three: for each coefficients set in level r, compute the variance among its own coefficients, select a coefficients set with the lowest coefficient variance, consider this coefficients set as F_r and use it to represent all coefficients sets in level r.

Step four: Construct coefficients set F_r' though choosing the top p percent of Fourier coefficients from F_r .

Output F_r'as the compressed and denoised image

Decoding:

Step five: apply invert Fourier transform to F_r' to get the reconstructed boundary I_r' .

$$I_r'(k_r) = \frac{1}{(p \times N_r)} \sum_{u_r=0}^{(p\times N_r)-1} F_r'(u_r) e^{\frac{j2\pi u_r k_r}{(p\times N_r)}}$$

Where $k_r = 0,1,2, \dots, N_r - 1; N_r = \frac{N}{2^r}; m = 1, 2, 3, \dots, 2^r; r = 0, 1, 2, \dots, R; j = \sqrt{-1}$

. Step six: compute the decompressed and denoised image boundary S_r' by resizing the reconstructed boundary I_r' based on interpolation to the original boundary size.

Output S_r' as decompressed and denoised image.

Now we take a simulation to show the effectiveness of our image compression and denoising algorithm based on multi-resolution Fourier.

The wavelets transform is the most widely used method that can do image compression and denoising simultaneously. Therefore our simulation is to compare our algorithm with 1D wavelets transform in term of the computation accuracy and efficiency.

(1) Simulation Data

We use 70 boundaries from standard image database ‘Mpeg7’, that is carefully designed to include most challenges in shape analysis [162], as our simulation data. The samples of data are shown in Figure 5-1.

We add Gaussian and ‘Salt &Pepper” noise into the original image data respectively. Then we apply our algorithm and wavelets transform to these image data corrupted by noise to do image compression and denoising simultaneously.

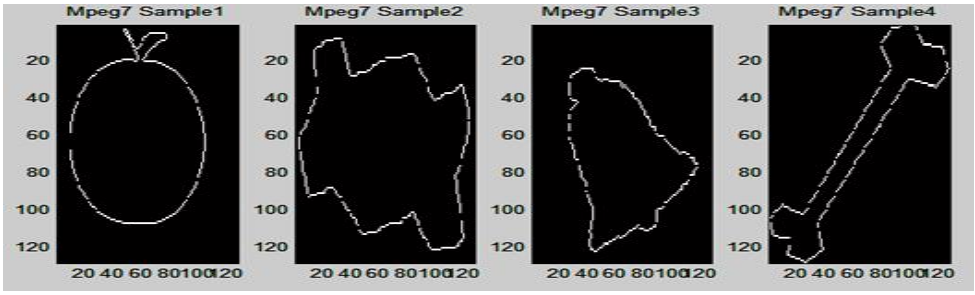

Figure 5.1 Samples of 'Mpeg7'

(2) Simulation Process

(i) *Implement our image compression and denoising algorithm based on multi-resolution Fourier descriptors.*

We construct the multi-resolution Fourier F_r^m by applying the traditional Fourier transform to the odd-even image boundary tree S_r^m where $m = 1, 2, 3, \dots, 2^r; r = 0, 1, 2.$

We choose the Fourier coefficients set with the lowest coefficients variance among its siblings; therefore we get F_1 and F_2 for level one and level two respectively.

We get F_1' and F_2' by respectively extracting top 75% coefficients from F_1 and F_2; F_1' and F_2' are the compressed & denoised images with compression ratio $a_1 > 2$ and $a_2 > 4$ respectively.

We apply invert Fourier to F_1' and F_2' to get the reconstructed image boundary I_1' and I_2' respectively; then we compute the decompressed & denoised image $S_1'$and S_2' through respectively resizing I_1' and I_2' to the original image size based on interpolation.

(ii) *Implement wavelet transform based on Haar and Biorthogobal 1.5 wavelet respectively*

Haar [127, 150] and Biorthogonal 1.5 [163-164] have been widely used in image compression and dinoising; in our simulation we respectively implement these two kinds of

wavelet transform in two levels. In each level, we consider the approximation coefficients with the threshold details as the compressed image. We use Matlab function $ddencmp('cmp','wp',X)$ where cmp is for compression, wp is for wavelet, X is the input data, to get the threshold. Therefore the compression ratio of wavelets in level one and level two are $a_1 < 2$ and $a_2 < 4$ respectively.

We use two popular measurements to demonstrate the computation accuracy.

One is root mean square errors (RMSE) [165] between the original image without noise and the decompressed & denoised image. The lower RMSE is, the higher quality of the decompressed & denoised image is.

Another is peak signal-to-noise ratio (PSNR) [166] between the original image without noise and the decompressed & denoised image. The higher PSNR is, the higher quality of the decompressed & denoised image is.

We also use execute time for each method to measure the computation efficiency.

(3) Simulation Results

Simulation results are shown in Table 5.1 -5.3 and Figure 5.2-5.7.

In these tables and figures, H_1 and H_2 represent the algorithm based on Haar wavelets in level one and level two respectively; B_1 and B_2 represent the algorithm based on Biorthogonal 1.5 wavelets in level one and level two respectively; F_1 and F_2 represent our algorithm based on multi-resolution Fourier in level one and level two respectively.

Table 5-1 Average RMSE of Multi-resolution Fourier and Wavelets

Algorithm	Average RMSE		Ranking of Average RMSE	
	Gaussian	Salt & Pepper	Gaussian	Salt & Pepper
$H1$	0.0285797	0.0144431	5	3
$H2$	0.0208413	0.014854	3	5
$B1$	0.0286909	0.0144937	6	4
$B2$	0.0208915	0.0149878	4	6
$F1$	**0.0197817**	**0.0138218**	**2**	**2**
$F2$	**0.0135818**	**0.0105157**	**1**	**1**

Table 5-2 Average PSNR of Multi-resolution Fourier and Wavelets

Algorithm	Average PSNR		Ranking of Average PSNR	
	Gaussian	Salt & Pepper	Gaussian	Salt & Pepper
$H1$	15.4713	18.606	2	4
$H2$	16.8548	18.4232	4	2
$B1$	15.454	18.5926	1	3
$B2$	16.8442	18.3845	3	1
$F1$	**17.0787**	**18.7088**	**5**	**5**
$F2$	**18.7363**	**19.9426**	**6**	**6**

Table 5-3 Average Computation Time of Multi-resolution Fourier and Wavelets

Algorithm	Average Time		Ranking of Average Time	
	Gaussian	Salt & Pepper	Gaussian	Salt & Pepper
$H1$	0.0925666	0.115869	5	5
$H2$	0.0950609	0.118815	6	6
$B1$	0.0885616	0.111391	3	3
$B2$	0.0922886	0.11557	4	4
$F1$	**0.0876314**	**0.110809**	**2**	**2**
$F2$	**0.0865661**	**0.109631**	**1**	**1**

Table 5.1 shows average RMSE generated by our algorithm, Haar wavelet, and Biorthogobal 1.5 wavelet under Gaussian and "Salt & Pepper" noise respectively. In the same

level, our algorithm always generates the lowest average RMSE that is our algorithm has the highest computation accuracy in term of RMSE.

Table 5.2 shows average PSNR generated by our algorithm, Haar wavelet, and Biorthogobal 1.5 wavelet under Gaussian and “Salt & Pepper” noise respectively. In the same level, our algorithm always generates the highest average PSNR, that is our algorithm has the highest computation accuracy in term of PSNR.

Table 5.3 shows the average computation time for our algorithm, Haar wavelet, and Biorthogobal 1.5 wavelet under Gaussian and “Salt & Pepper” noise respectively. In the same level, our algorithm always generates the lowest average computation time,that is our algorithm has the highest computation efficiency.

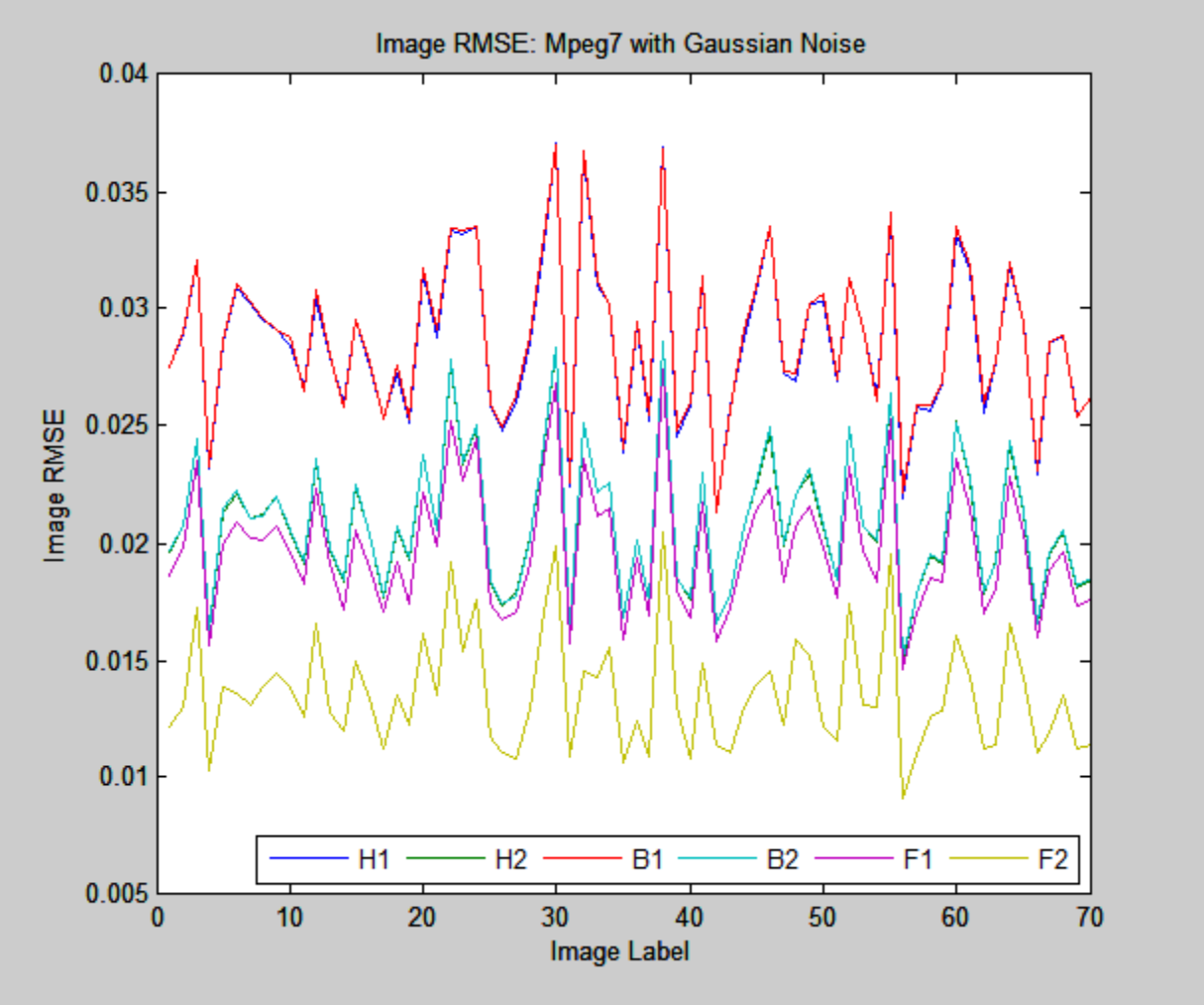

Figure 5.2 RMSE of Multi-resolution Fourier and Wavelets under Gaussian Noise for 'Mpeg7'

Figure 5.2 shows the RMSE respectively generated by our algorithm, Haar wavelet, and Biorthogobal 1.5 wavelet under Gaussian noise. At the bottom of this figure, there are two lines: a yellow line and a purple line that respectively represent RMSE generated by our algorithm in level two and level one. Obviously, our algorithm has the highest computation accuracy in term of RMSE.

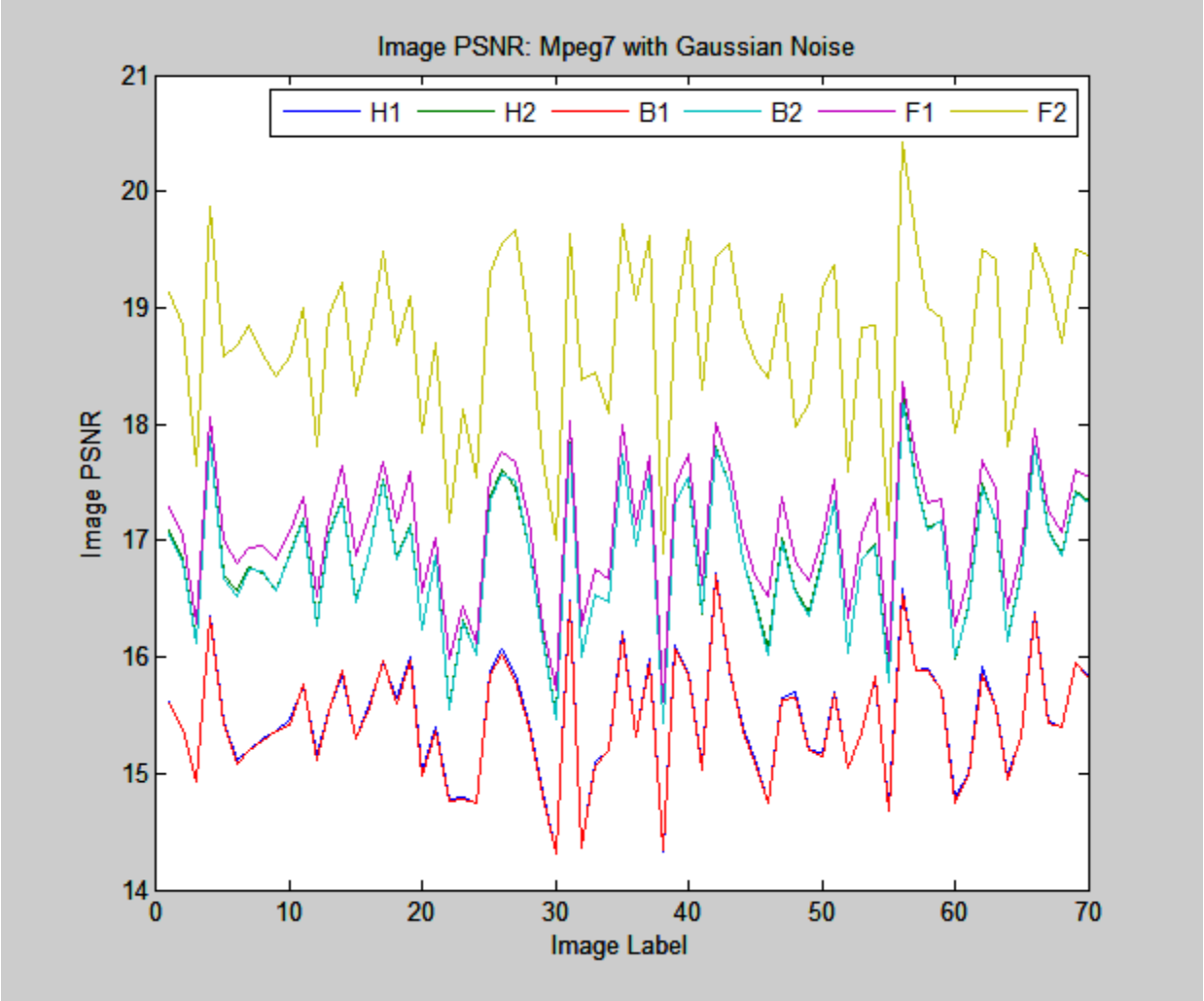

Figure 5.3 PSNR of Multi-resolution Fourier and Wavelets under Gaussian Noise for 'Mpeg7'

Figure 5.3 shows the PSNR respectively generated by our algorithm, Haar wavelet, and Biorthogobal 1.5 wavelet under Gaussian noise. At the top of this figure, there are two lines: a yellow line and a purple line that respectively represent PSNR generated by our algorithm in level two and level one. Obviously, our algorithm has the highest computation accuracy in term of PSNR.

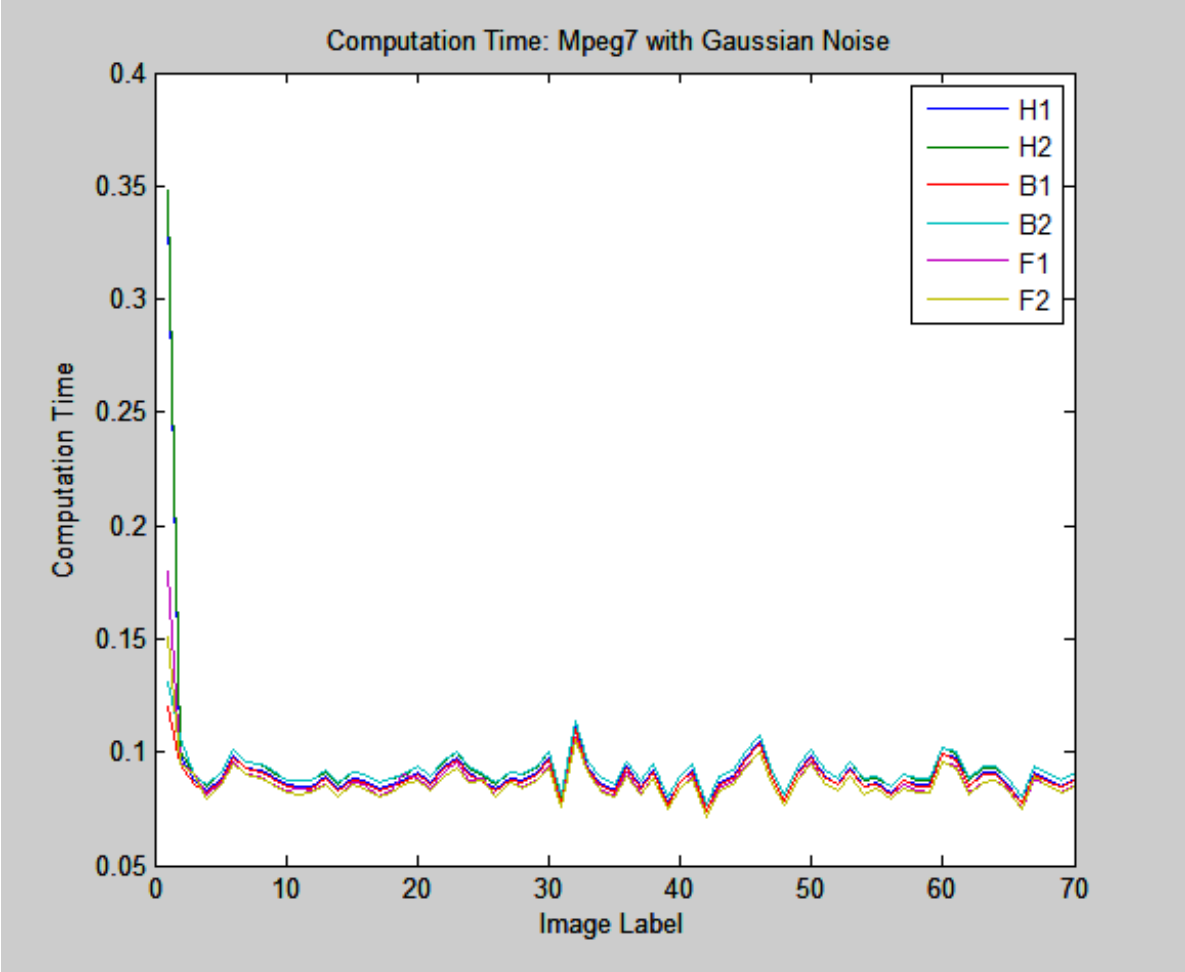

Figure 5.4 Computation Time of Multi-resolution Fourier and Wavelets under Gaussian Noise for 'Mpeg7'

Figure 5.4 shows computation time respectively generated by our algorithm, Haar wavelet, and Biorthogobal 1.5 wavelet under Gaussian noise. At the bottom of this figure, there are two lines: a yellow line and a purple line, respectively representing the RMSE generated by our algorithm in level two and level one. Obviously, our algorithm has the highest computation efficiency.

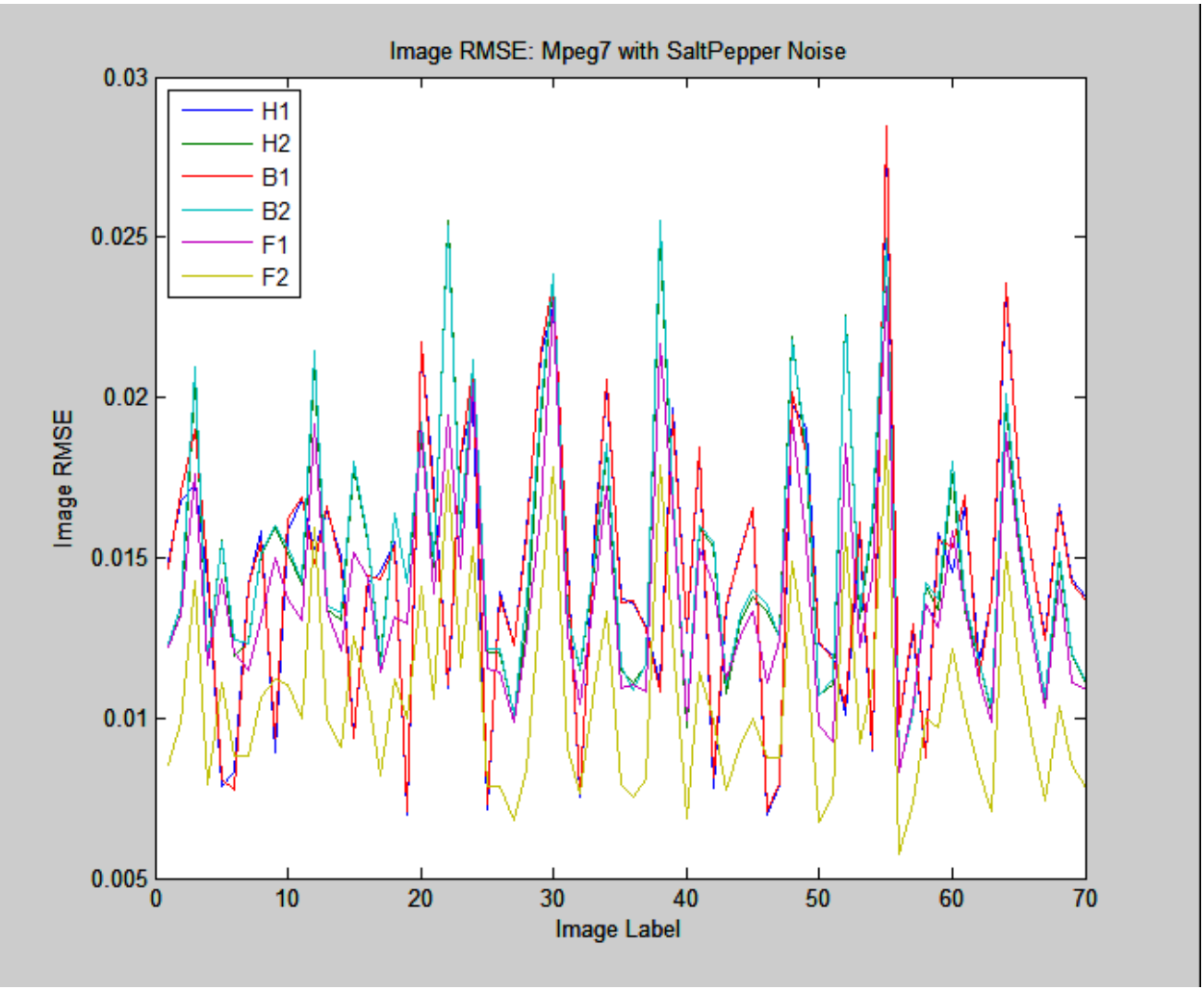

Figure 5.5 RMSE of Multi-resolution Fourier and Wavelets under 'Salt & Pepper' Noise for 'Mpeg7'

Figure 5.5 shows the RMSE respectively generated by our algorithm, Haar wavelet, and Biorthogobal 1.5 wavelet under 'Salt &Pepper' noise. At the bottom of this figure, there are three lines, a yellow line, representing RMSE generated by our algorithm in level two, a purple line, representing RMSE generated by our algorithm in level one, and a red line representing RMSE generated by Biorthogobal 1.5 wavelet in level one. The yellow line is usually below the purple and red line; the purple line is usually between the yellow line and red line. Obviously, in the same level, our algorithm generates the lowest RMSE, that is our algorithm has the highest computation accuracy in term of RMSE.

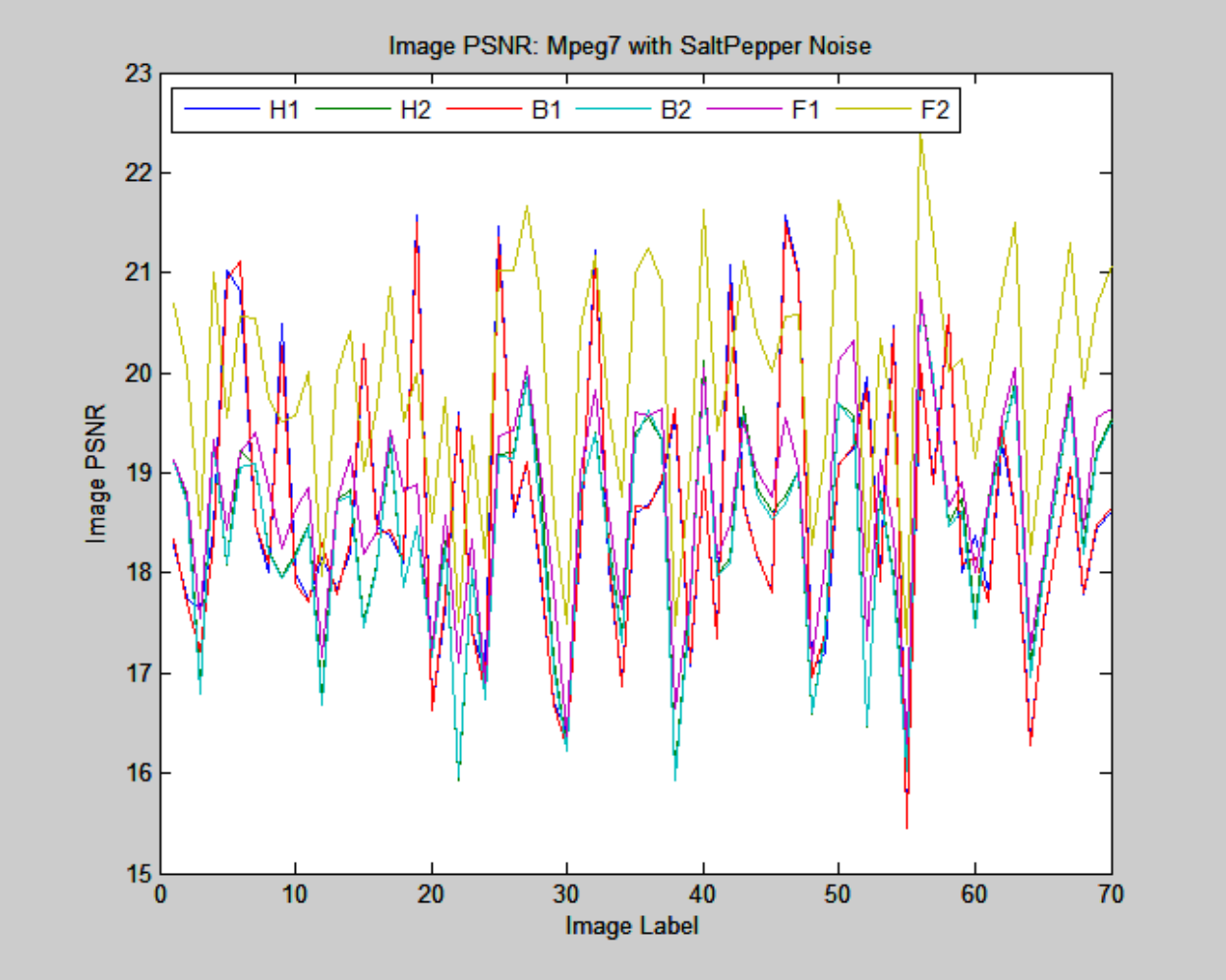

Figure 5.6 PSNR of Multi-resolution Fourier and Wavelets under 'Salt & Pepper' Noise for 'Mpeg7'

Figure 5.6 shows the PSNR respectively generated by our algorithm, Haar wavelet, and Biorthogobal 1.5 wavelet under 'Salt &Pepper' noise. At the top of this figure, there are three lines, a yellow line, representing PSNR generated by our algorithm in level two, a purple line, representing PSNR generated by our algorithm in level one, and a red line representing PSNR generated by Biorthogobal 1.5 wavelet in level one. The yellow line is usually above the purple and red line; the purple line is usually between the yellow line and red line. Obviously, in the same level, our algorithm generates the highest PSNR, that is our algorithm has the highest computation accuracy in term of PSNR.

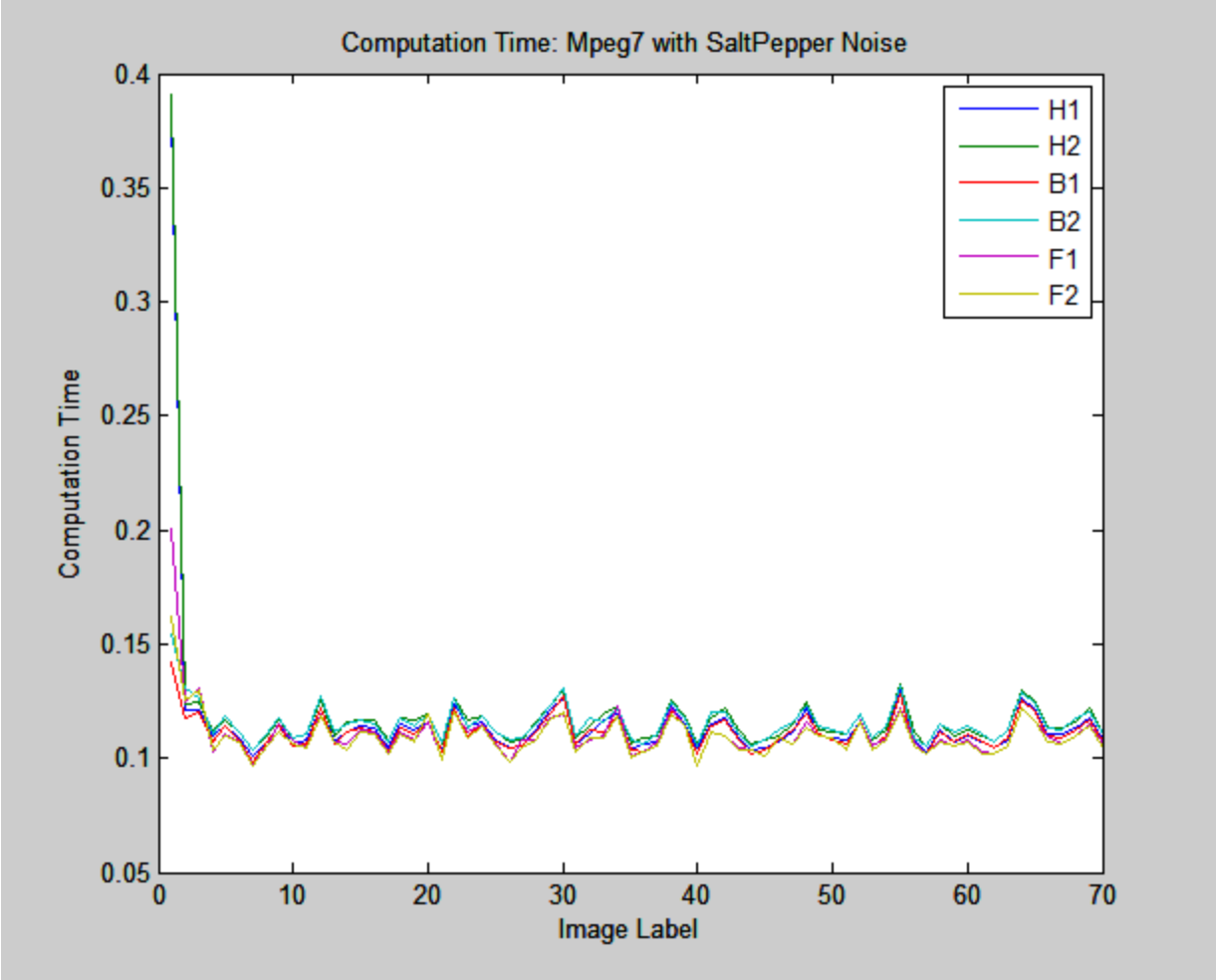

Figure 5.7 Computation Time of Multi-resolution Fourier and Wavelets under 'Salt & Pepper' Noise for 'Mpeg7'

Figure 5.7 shows the computation time respectively generated by our algorithm, Haar wavelet, and Biorthogobal 1.5 wavelet under 'Salt &Pepper' noise. At the bottom of this figure, there are two lines: a yellow line and a purple line, respectively representing computation time generated by our algorithm in level two and level one. Obviously, our algorithm has the highest computation efficiency.

The simulation results clearly demonstrate that our algorithm has higher compression accuracy and efficiency than wavelets transform.

5.2.2 Image Compression and Denoising Algorithm based on Multi-resolution Discrete Cosine Transform

Discrete cosine transform (DCT) has excellent energy compaction property; however the performance of its traditional coder generally degrades at high compression ratios mainly due to the underlying block-based strategy. Usually, wavelets have higher computation accuracy than DCT in the area of image compression. We propose a novel image compression and denoising algorithm based on multi-resolution DCT. Our algorithm eliminates the blocking artifact and outperforms wavelets in terms of computation accuracy and efficiency.

Algorithm V: Image Compression and Denoising based on Multi-resolution Discrete Cosine Transform (DCT)

Input: Image $H(k_1, k_2)$ ($k_1 = 0, 1, 2, \dots, K_1 - 1$ and $k_2 = 0, 1, 2, \dots, K_2 - 1$)

Output: compressed and denoised image B_r'

decompressed and denosied image H_r'

Encoding:

Step one: Transform image H into an odd-even image region tree H_r^m.

At node m of tree level r, the multi-resolution image H_r^m is represented as:

$$H_r^m((k_1)_r^m, (k_2)_r^m)$$

$$where\ (k_1)_r^m = 0, 1, 2, \dots (K_1) - 1;\ (K_1)_r = \frac{K_1}{2^r}\ ;$$

$$(k_2)_r^m = 0, 1, 2, \dots (K_2)_r - 1 \colon (K_2)_r = \frac{K_2}{2^r};$$

$$m = 1, 2, 3, \dots, 4^r; r = 0,1,2, \dots, R.$$

Step two: get multi-resolution DCT B_r^m , by applying traditional DCT to the image tree .

$$B_r^m(u_r^m, v_r^m)$$

$$= w_1(u_r^m) \times w_2(v_r^m)$$

$$\times \sum_{(k_1)_r^m=0}^{(K_1)_r-1} \sum_{(k_2)_r^m=0}^{(K_2)_r-1} H_r^m((k_1)_r^m, (k_2)_r^m) \times cos\frac{(2 \times (k_1)_r^m + 1) \times u_r^m \times \pi}{2 \times (K_1)_r}$$

$$\times cos\frac{(2 \times (k_2)_r^m + 1) \times v_r^m \times \pi}{2 \times (K_2)_r} \quad (5.1)$$

Where $w_1(u_r^m) = \begin{cases} \frac{1}{\sqrt{(K_1)_r}} & u_r^m = 0 \\ \sqrt{\frac{2}{(K_1)_r}} & u_r^m = 1,2, \dots, (K_1)_r - 1 \end{cases}$

$$w_2(v_r^m) = \begin{cases} \frac{1}{\sqrt{(K_2)_r}} & v_r^m = 0 \\ \sqrt{\frac{2}{(K_2)_r}} & v_r^m = 1,2, \dots, (K_2)_r - 1 \end{cases}$$

$$u_r^m = 0, 1, 2, \dots, (K_1)_r - 1;\ v_r^m = 0, 1, 2, \dots, (K_2)_r - 1;$$

$$m = 1, 2, 3, \dots, 4^r;\ r = 0,1,2, \dots, R$$

Step three: compute mean of the DCT coefficients sets in level r

$$B_r(u_r, v_r) = \frac{1}{4^r} \sum_{m=1}^{4^r} B_r^m(u_r^m, v_r^m) \quad (5.2)$$

Where $u_r = 0, 1, 2, \dots, (K_1)_r - 1;\ v_r = 0, 1, 2, \dots, (K_2)_r - 1.$

Step four: compute B_r' by doing hard threshold to B_r, based on a threshold T.

Output B_r' as the compressed and denoised image.

Decoding:

Step five: Generate reconstructed image I_r', through applying invert DCT to B_r'.

$$I_r'((k_1)_r,(k_2)_r)$$

$$= \sum_{u_r=0}^{(K_1)_r-1} \sum_{v_r=0}^{(K_2)_r-1} w_1(u_r) \times w_2(v_r) \times B_r'(u_r,v_r)$$

$$\times cos\frac{(2\times(k_1)_r+1)\times u_r\times\pi}{2\times(K_1)_r} \times cos\frac{(2\times(k_2)_r+1)\times v_r\times\pi}{2\times(K_2)_r} \tag{5.3}$$

$$Where \quad w_1(u_r^m) = \begin{cases} \frac{1}{\sqrt{(K_1)_r}} & u_r^m = 0 \\ \sqrt{\frac{2}{(K_1)_r}} & u_r^m = 1,2,\dots,(K_1)_r-1 \end{cases}$$

$$w_2(v_r^m) = \begin{cases} \frac{1}{\sqrt{(K_2)_r}} & v_r^m = 0 \\ \sqrt{\frac{2}{(K_2)_r}} & v_r^m = 1,2,\dots,(K_2)_r-1 \end{cases}$$

$$(k_1)_r = 0,1,2,\dots,(K_1)_r-1;\ (k_2)_r = 0,1,2,\dots,(K_2)_r-1.$$

Step six: Generate decompressed and denoised image H_r', by resizing I_r' to the original image size based on bicubic interpolation.

Output H_r' as the decompressed and denoised image.

Now we take simulation to show the effectiveness of our image compression and denoising algorithm based on multi-resolution DCT. The wavelet transform is the most widely used method that can do image compression and denoising simultaneously. The simulation is to compare our algorithm with 2D wavelet transform in term of the computation accuracy and efficiency.

(1) Simulation Data

We use two image databases.

(i) *Forests*

This database has 328 images about different forests. Its format is jpg. It is downloaded from http://www-cvr.ai.uiuc.edu/ponce_grp/data/ (under the "Fifteen Scene Categories" of the webpage). The samples of this database are shown in Figure 5.8.

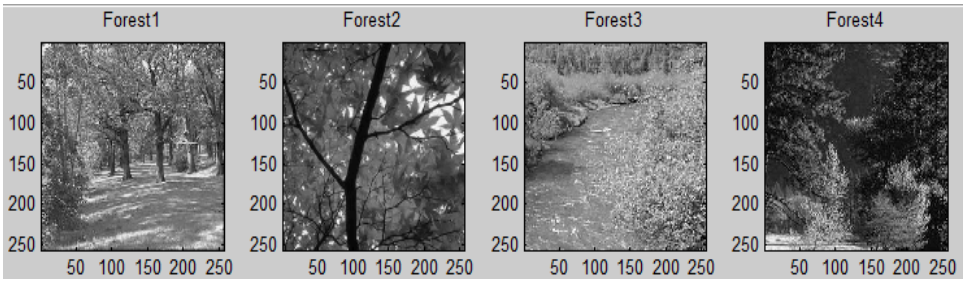

Figure 5.8 Samples of 'Forest'

(ii) *PCA*

This image database has 91 images about different backgrounds, such as people, offices, mountains, woods, cars, flowers, roads, river, etc. Its image format is gif. It is downloaded from http://pics.psych.stir.ac.uk/Other_image_types.htm. The samples of this database are shown in Figure 5.9.

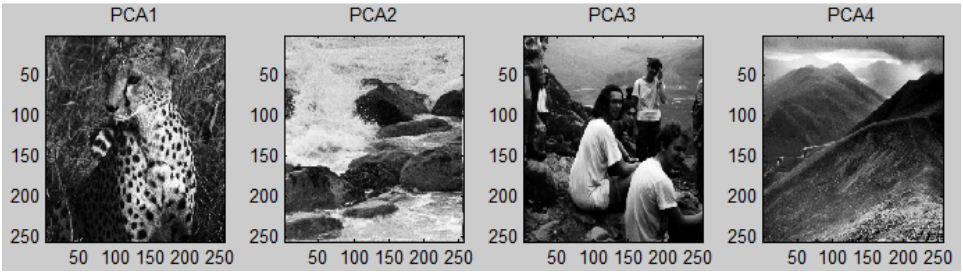

Figure 5.9 Samples of 'PCA'

We add Gaussian and 'Salt &Pepper" noise into the original image data respectively. Then we apply our algorithm and wavelets transform to these image data corrupted by noise for image compression and denoising.

(2) Simulation Process

(i) *Implement our image compression and denoising algorithm based on multi-resolution discrete cosine transform (DCT).*

We construct the multi-resolution DCT B_r^m by applying the traditional DCT to the odd-even image region tree H_r^m where $m = 1, 2, 3, \dots, 4^r; r = 0, 1, 2.$

We compute the mean of the coefficients tree nodes from the same level: we get $B_1 = \frac{1}{4}\sum_{m=1}^{4} B_1^m$ from level one and $B_2 = \frac{1}{16}\sum_{m=1}^{16} B_2^m$ from level two.

We get B_1' and B_2' by respectively doing hard thresholding to B_1 and B_2 with threshold $T = 0.01$; B_1' and B_2' are the compressed & denoised images with compression ratio $a_1 > 4$ and $a_2 > 16$ respectively.

We respectively apply invert DCT to B_1' and B_2' to get the reconstructed image I_1' and I_2'; then we compute the decompressed & denoised image $H_1'$and H_2' through respectively resizingI_1' and I_2' to the original image size based on bicubic interpolation.

(ii) *Implement wavelet transform based on Haar and Biorthogobal 1.5 wavelet respectilvey*

Haar [127, 150] and Biorthogonal 1.5 [163-164] have been widely used in image compression and dinoising; in our simulation we respectively implement these two kinds of wavelet transform in two levels. In each level, we consider the approximation coefficients with the threshold details as the compressed image. We use Matlab function $ddencmp('den','wp',X)$ where den is for denoising, wp is for wavelet, X is the input data, to get the threshold. Therefore the compression ratio of wavelets in level one and level two are $a_1 < 4$ and $a_2 < 16$ respectively.

We use two popular measurements to demonstrate the computation accuracy.

One is root mean square errors (RMSE) [165] between the original image without noise and the decompressed & denoised image. The lower RMSE is, the higher quality of the decompressed & denoised image is.

Another is peak signal-to-noise ratio (PSNR) [166] between the original image without noise and the decompressed & denoised image. The higher PSNR is, the higher quality of the decompressed & denoised image is.

We also use execute time for each method to measure the computation efficiency.

(3) Simulation Results

Simulation Results are shown in Table 5.4 -5.6 and Figure 5.10-5.21.

In these tables and figures, H_1 and H_2 represent the algorithm based on Haar wavelets in level one and level two respectively; B_1 and B_2 represent the algorithm based on Biorthogonal 1.5 wavelets in level one and level two respectively; DCT_1 and DCT_2 represent our algorithm based on multi-resolution DCT in level one and level two respectively.

Table 5.4 compares our algorithm with Haar and Biorthogobal 1.5 wavelets in term of average RMSE under Gaussian and "Salt & Pepper" noise respectively. In the same level, our algorithm always generates the lowest average RMSE, ,that is our algorithm has the highest computation accuracy in term of RMSE.

Table 5.5 compares our algorithm with Haar and Biorthogobal 1.5 wavelets in term of average PSNR under Gaussian and "Salt & Pepper" noise respectively. In the same level, our algorithm always generates the highest average PSNR, that is our algorithm has the highest computation accuracy in term of PSNR.

Table 5.6 compares our algorithm with Haar and Biorthogobal 1.5 wavelets in term of average computation time under Gaussian and "Salt & Pepper" noise respectively. In the same level, our algorithm always generates the lowest average computation, that is our algorithm has the highest computation efficiency.

Table 5-4 Average RMSE of Multi-resolution DCT and Wavelets

Data Set	Algorithm	Average RMSE		Ranking of Average RMSE	
		Gaussian	Salt & Pepper	Gaussian	Salt & Pepper
Forest	$H1$	0.00958405	0.0113352	2	2
	$H2$	0.0136558	0.0141522	5	5
	$B1$	0.0102577	0.0121408	3	3
	$B2$	0.0142929	0.0147958	6	6
	$\boldsymbol{DCT1}$	**0.00824485**	**0.00948981**	**1**	**1**
	$\boldsymbol{DCT2}$	**0.0126351**	**0.0130464**	**4**	**4**
PCA	$H1$	0.00522994	0.00769219	2	3
	$H2$	0.00728058	0.0079566	5	4
	$B1$	0.00562845	0.00834689	3	5
	$B2$	0.00771076	0.00846867	6	6
	$\boldsymbol{DCT1}$	**0.00397602**	**0.00565023**	**1**	**1**
	$\boldsymbol{DCT2}$	**0.0060321**	**0.00653967**	**4**	**2**

Table 5-5 Average PSNR of Multi-resolution DCT and Wavelets

Data Set	Algorithm	Average PSNR		Ranking of Average PSNR	
		Gaussian	Salt & Pepper	Gaussian	Salt & Pepper
Forest	$H1$	20.6678	19.8261	5	5
	$H2$	19.2979	19.0978	2	2
	$B1$	20.3499	19.5096	4	3
	$B2$	19.0766	18.8819	1	1
	$\boldsymbol{DCT1}$	**21.5004**	**20.7446**	**6**	**6**
	$\boldsymbol{DCT2}$	**19.7028**	**19.5177**	**3**	**4**
PCA	$H1$	23.2713	21.3609	5	2
	$H2$	22.216	21.6712	2	4
	$B1$	22.9347	20.9909	3	1
	$B2$	21.9403	21.3659	1	3
	$\boldsymbol{DCT1}$	**24.6547**	**22.8167**	**6**	**6**
	$\boldsymbol{DCT2}$	**23.2134**	**22.6883**	**4**	**5**

Table 5-6 Average Computation Time of Multi-resolution DCT and Wavelets

Data Set	Algorithm	Average Time		Ranking of Average Time	
		Gaussian	Salt & Pepper	Gaussian	Salt & Pepper
Forest	$H1$	0.019407	0.0196875	3	3
	$H2$	0.031194	0.0318039	5	5
	$B1$	0.0207148	0.0213732	4	4
	$B2$	0.0345788	0.0357681	6	6
	$\boldsymbol{DCT1}$	**0.00810591**	**0.00841515**	**2**	**2**
	$\boldsymbol{DCT2}$	**0.00679535**	**0.00696821**	**1**	**1**
PCA	$H1$	0.0216125	0.0215681	4	4
	$H2$	0.0337857	0.034048	5	5
	$B1$	0.0215579	0.0214529	3	3
	$B2$	0.0358938	0.0360872	6	6
	$\boldsymbol{DCT1}$	**0.00867435**	**0.00867636**	**2**	**2**
	$\boldsymbol{DCT2}$	**0.00722847**	**0.00728542**	**1**	**1**

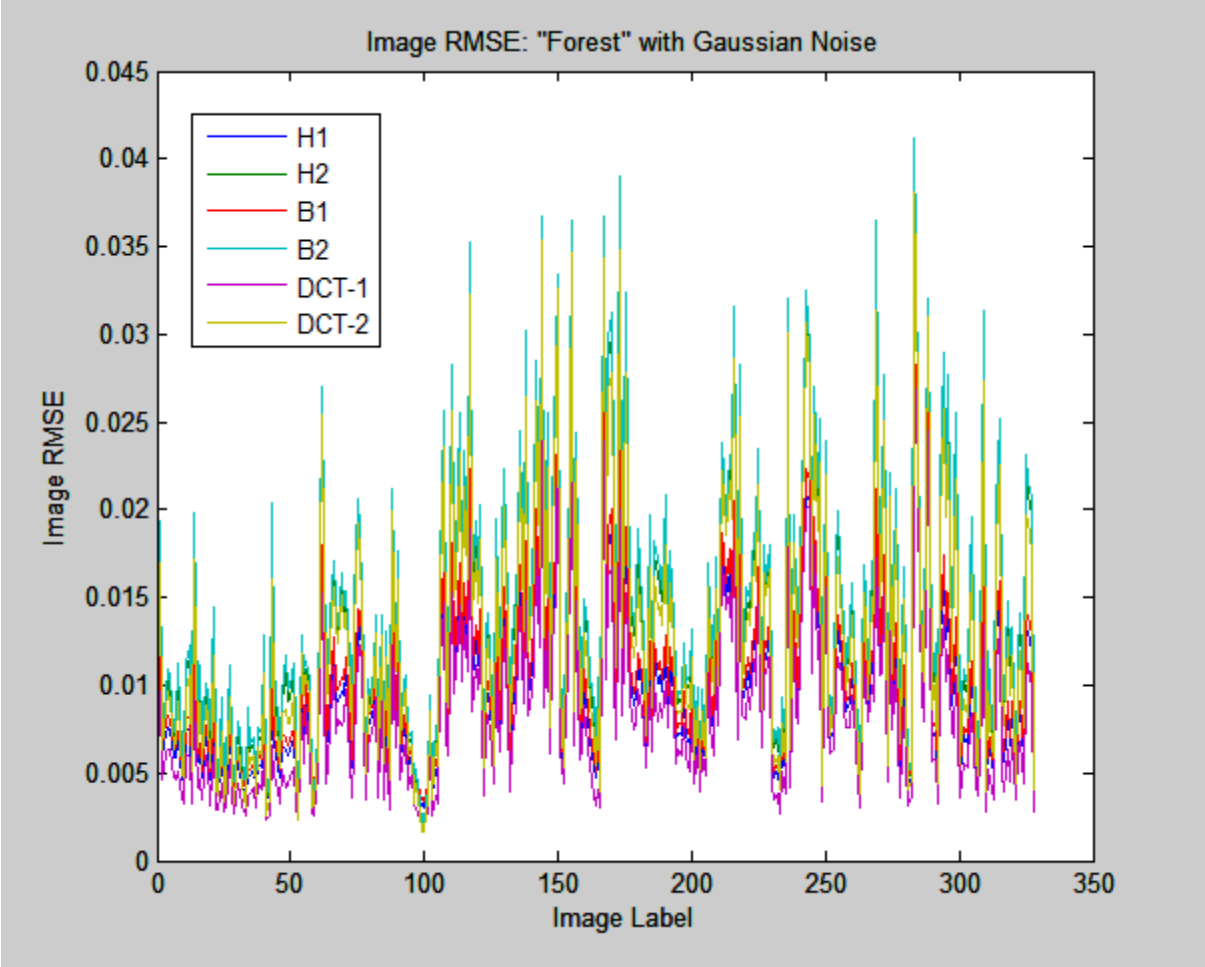

Figure 5.10 RMSE of Multi-resolution DCT and Wavelets under Gaussian Noise for 'Forest'

Figure 5.10 shows RMSE generated by our algorithm, Haar and Biorthogobal 1.5 wavelets under Gaussian noise for the dataset 'Forest'. At the bottom of this figure, there is a purple line representing the RMSE generated by our algorithm in level one. In the figure, there is a yellow line, green line and light blue line that respectively represent the RMSE generated by our algorithm, Haar and Biorthogobal 1.5 wavelets in level two. The green line and light blue line are usually above the yellow line. Therefore, in the same level, our algorithm always generates the lowest RMSE, that is our algorithm has the highest computation accuracy in term of RMSE.

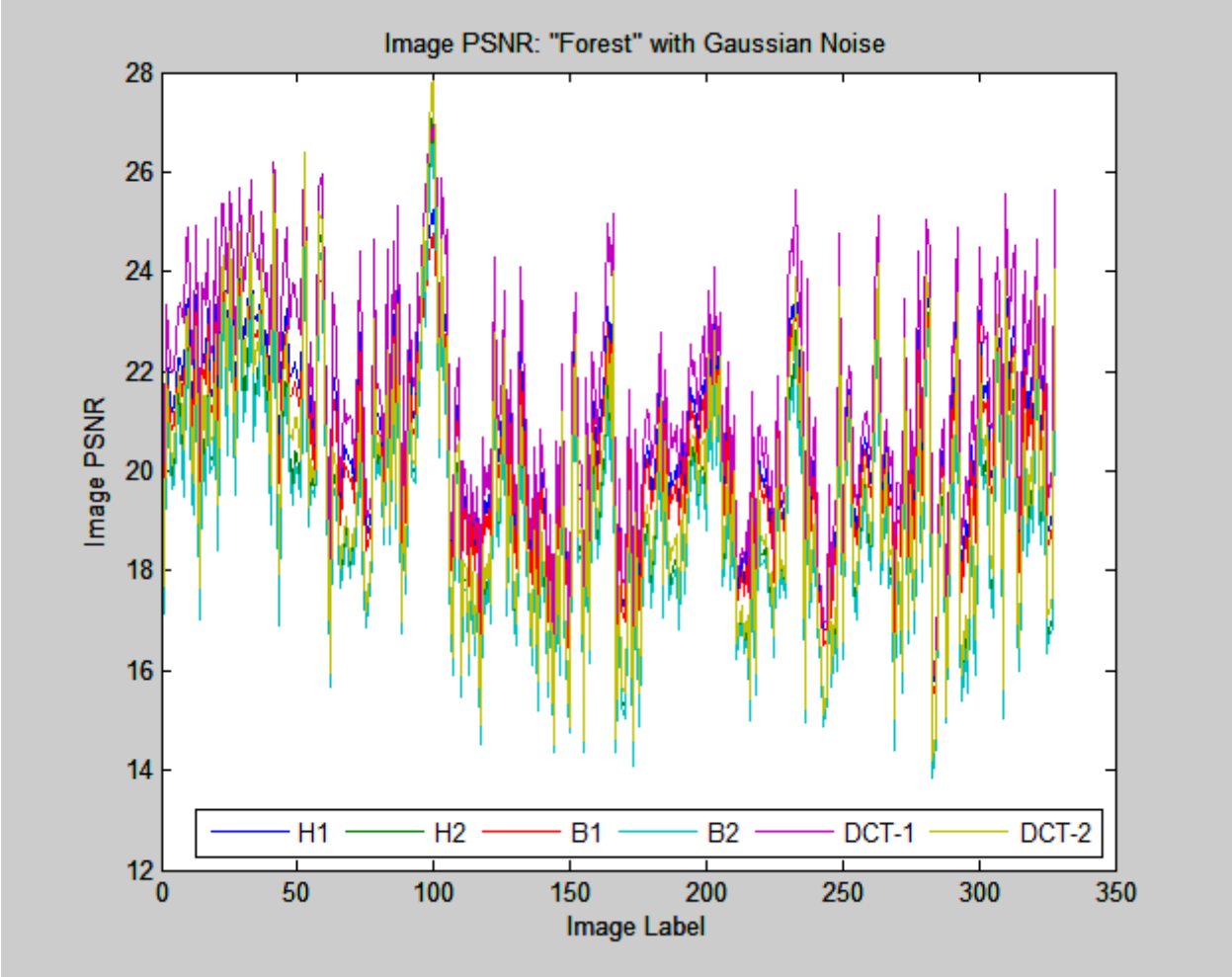

Figure 5.11 PSNR of Multi-resolution DCT and wavelets under Gaussian Noise for 'Forest'

Figure 5.11 shows PSNR generated by our algorithm, Haar and Biorthogobal 1.5 wavelets under Gaussian noise for the dataset 'Forest'. At the top of this figure, there is a purple line representing the PSNR generated by our algorithm in level one. In the figure, there is a yellow line, green line and light blue line that respectively represent the PSNR generated by our algorithm, Haar and Biorthogobal 1.5 wavelets in level two. The green line and light blue line are usually below the yellow line. Therefore, in the same level, our algorithm generates the highest PSNR, that is our algorithm has the highest computation accuracy in term of PSNR.

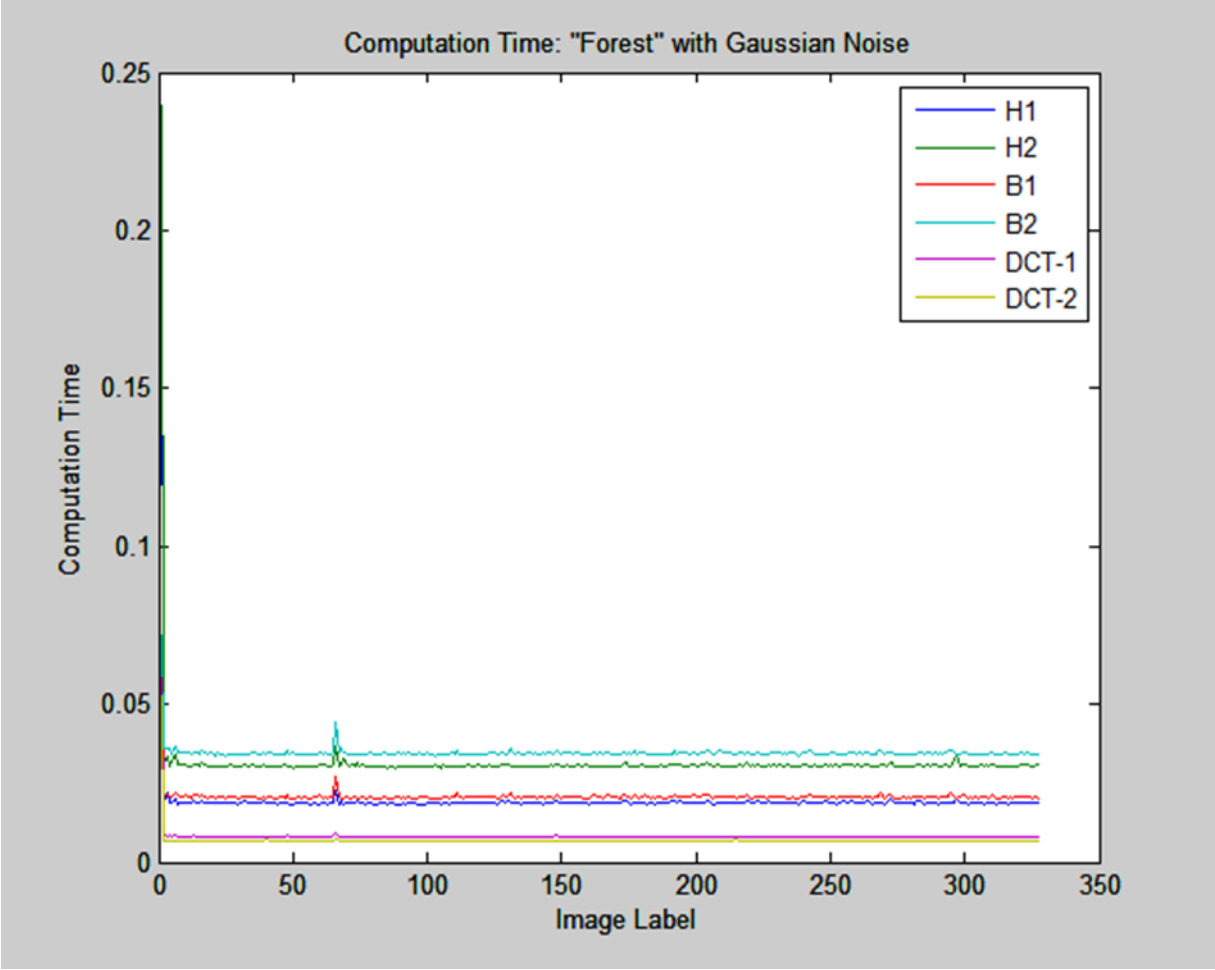

Figure 5.12 Computation Time of Multi-resolution DCT and Wavelets under Gaussian Noise for 'Forest'

Figure 5.12 shows computation time cost by our algorithm, Haar and Biorthogobal 1.5 wavelets under Gaussian noise for the dataset 'Forest'. At the bottom of this figure, there is a yellow line and a purple line, representing the computation time cost by our algorithm in level two and level one respectively. Therefore, our algorithm has the lowest computation time, that is our algorithm has the highest computation efficiency.

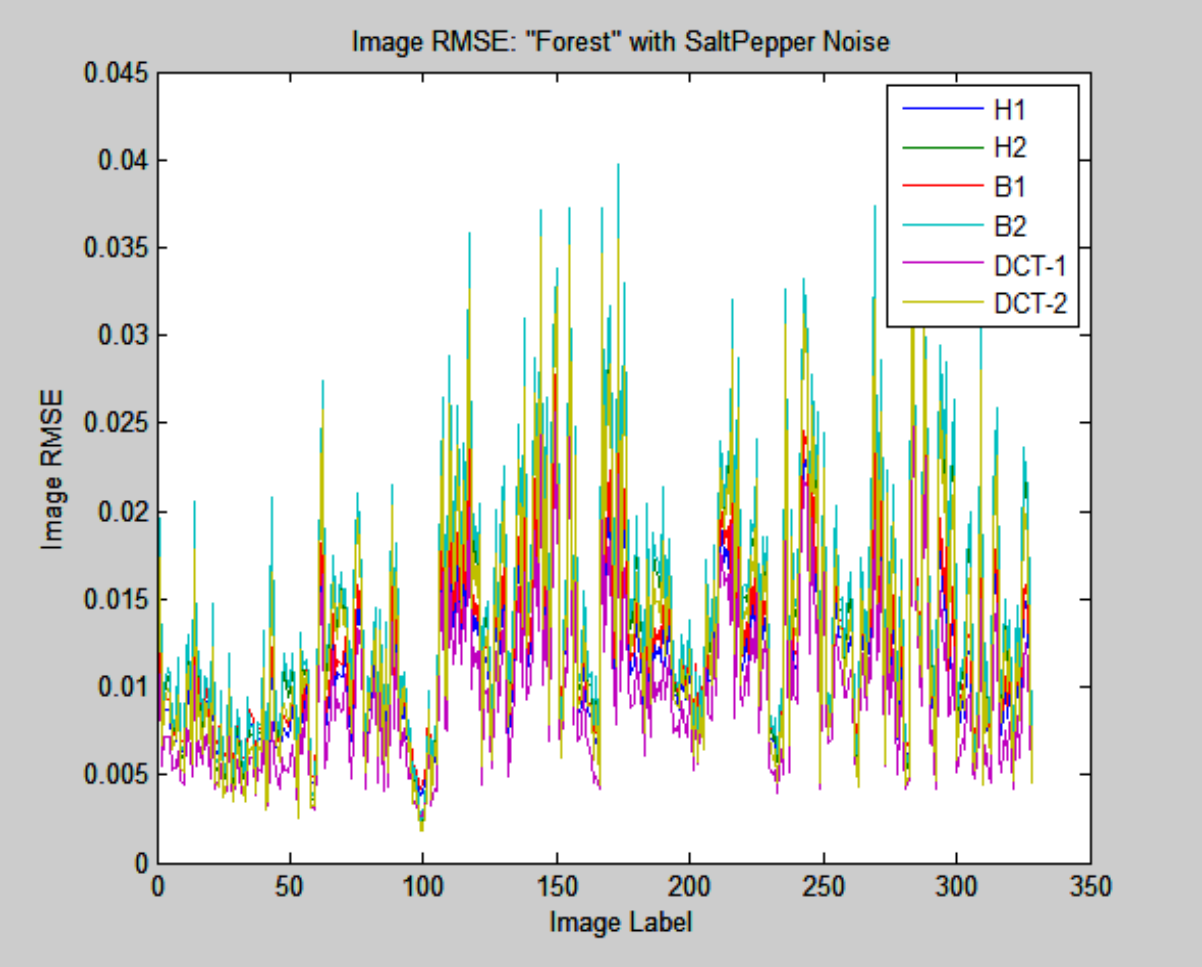

Figure 5.13 RMSE of Multi-resolution DCT and Wavelets under 'Salt & Pepper' Noise for 'Forest'

Figure 5.13 shows RMSE generated by our algorithm, Haar and Biorthogobal 1.5 wavelets under 'Salt & Pepper' noise for the dataset 'Forest'. There is a purple line and a yellow line, representing the RMSE generated by our algorithm in level one and level two respectively. The purple line is usually at the bottom of the figure, with some cases that the yellow line goes below the purple line. There is a green line and a light blue line, representing the RMSE generated by Haar and Biorthogobal 1.5 wavelets in level two respectively. The green line and a light blue line are usually above the yellow line. Therefore, in the same level, our algorithm

generates the lowest RMSE, that is our algorithm has the highest computation accuracy in term of RMSE.

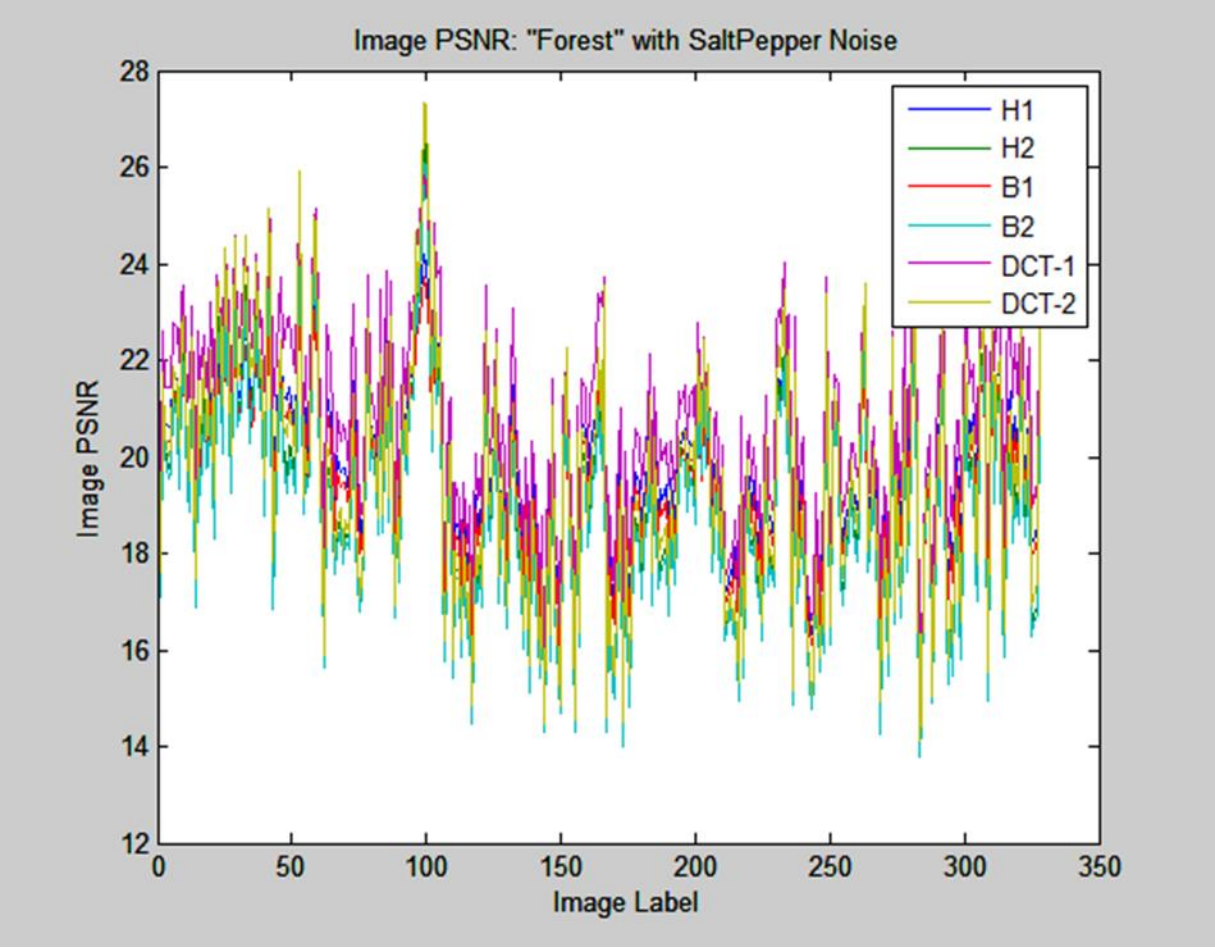

Figure 5.14 PSNR of Multi-resolution DCT and Wavelets under 'Salt & Pepper' Noise for 'Forest'

Figure 5.14 shows PSNR generated by our algorithm, Haar and Biorthogobal 1.5 wavelets under 'Salt & Pepper' noise for the dataset 'Forest'. There is a purple line and a yellow line, representing the RSNR generated by our algorithm in level one and level two respectively. The purple line is usually at the top of the figure, with some cases that the yellow line goes above the purple line. There is a green line and light blue line, representing the PSNR generated by Haar

and Biorthogobal 1.5 wavelets in level two respectively. The green line and light blue line are usually below the yellow line. Therefore, in the same level, our algorithm generates the highest PSNR, that is our algorithm has the highest computation accuracy in term of PSNR.

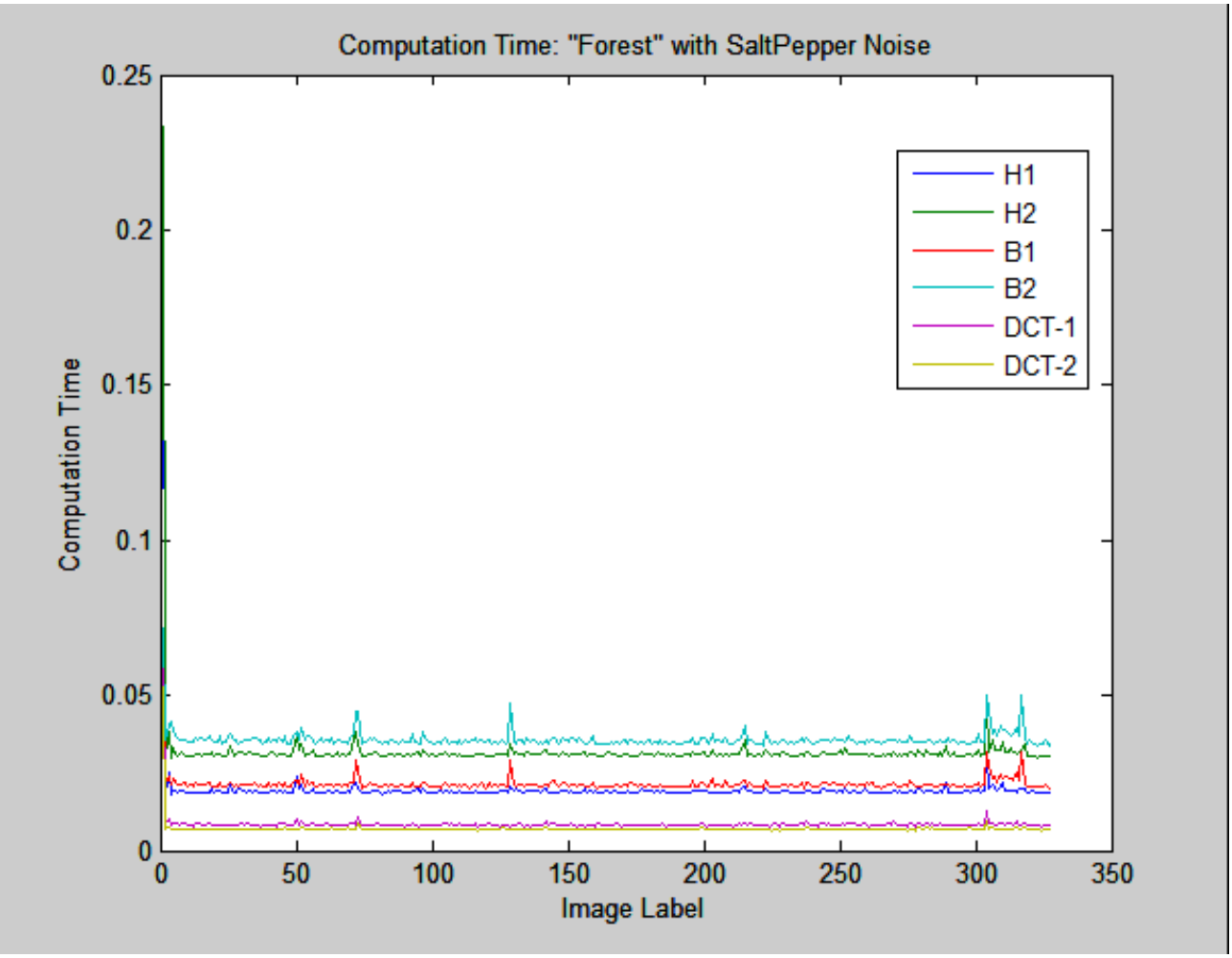

Figure 5.15 Computation Time of Multi-resolution DCT and Wavelets under 'Salt & Pepper' Noise for 'Forest'

Figure 5.15 shows computation time cost by our algorithm, Haar and Biorthogobal 1.5 wavelets under 'Salt & Pepper' noise for the dataset 'Forest'. At the bottom of this figure, there is a yellow line and a purple line, representing the computation time cost by our algorithm in

level two and one respectively. Therefore, our algorithm hast the lowest computation time, that is our algorithm has the highest computation efficiency.

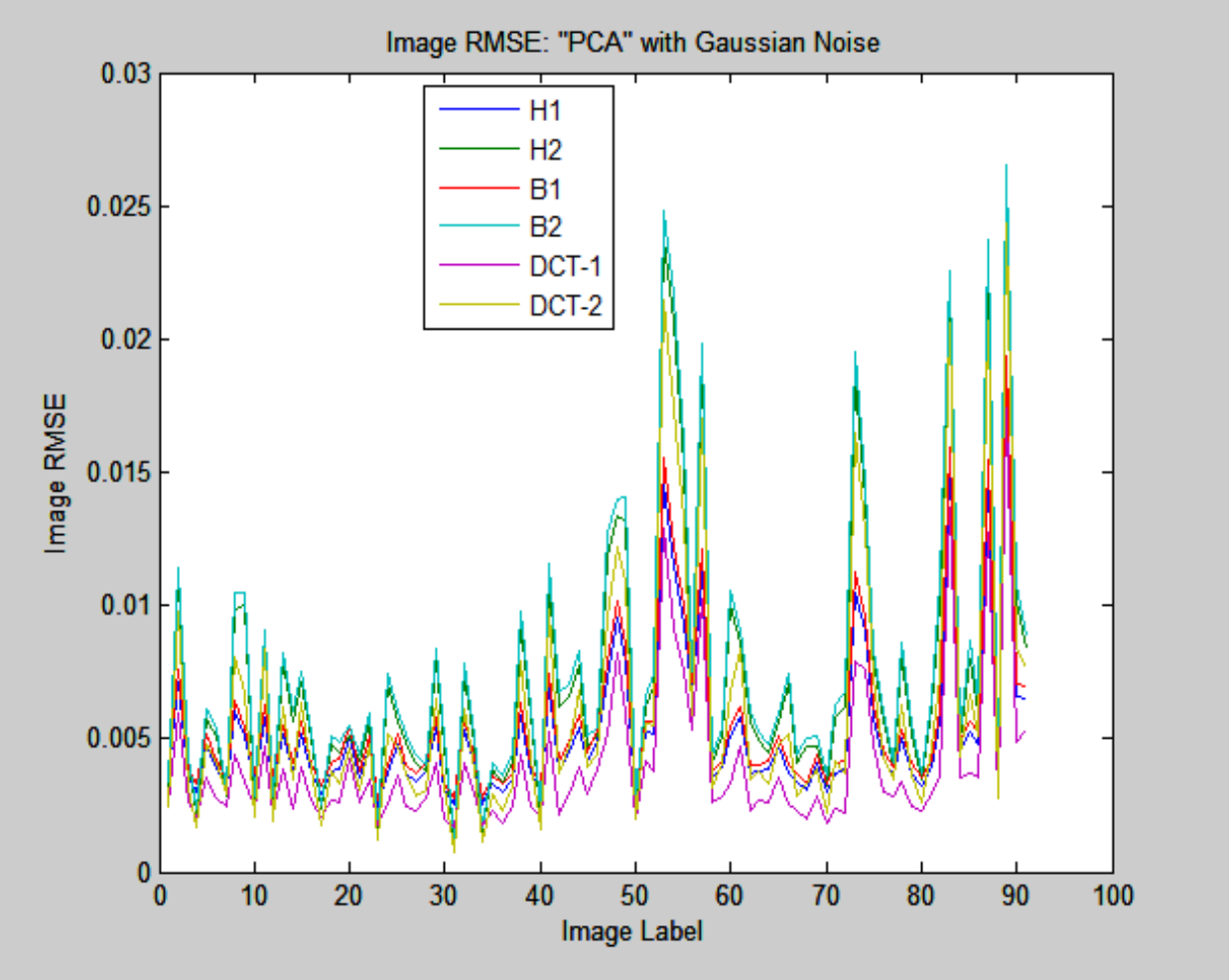

Figure 5.16 RMSE of Multi-resolution DCT and Wavelets under Gaussian Noise for 'PCA'

Figure 5.16 shows RMSE generated by our algorithm, Haar and Biorthogobal 1.5 wavelets under ‘Gaussian’ noise for the dataset ‘PCA’. There is a purple line and a yellow line, representing the RMSE generated by our algorithm in level one and level two respectively. The purple line is usually at the bottom of the figure, with some cases that the yellow line goes below the purple line. There is a green line and light blue line, representing the RMSE generated by Haar and Biorthogobal 1.5 wavelets in level two respectively. The green line and light blue line

are usually above the yellow line. Therefore, in the same level, our algorithm always generates the lowest RMSE, that is our algorithm has the highest computation accuracy in term of RMSE.

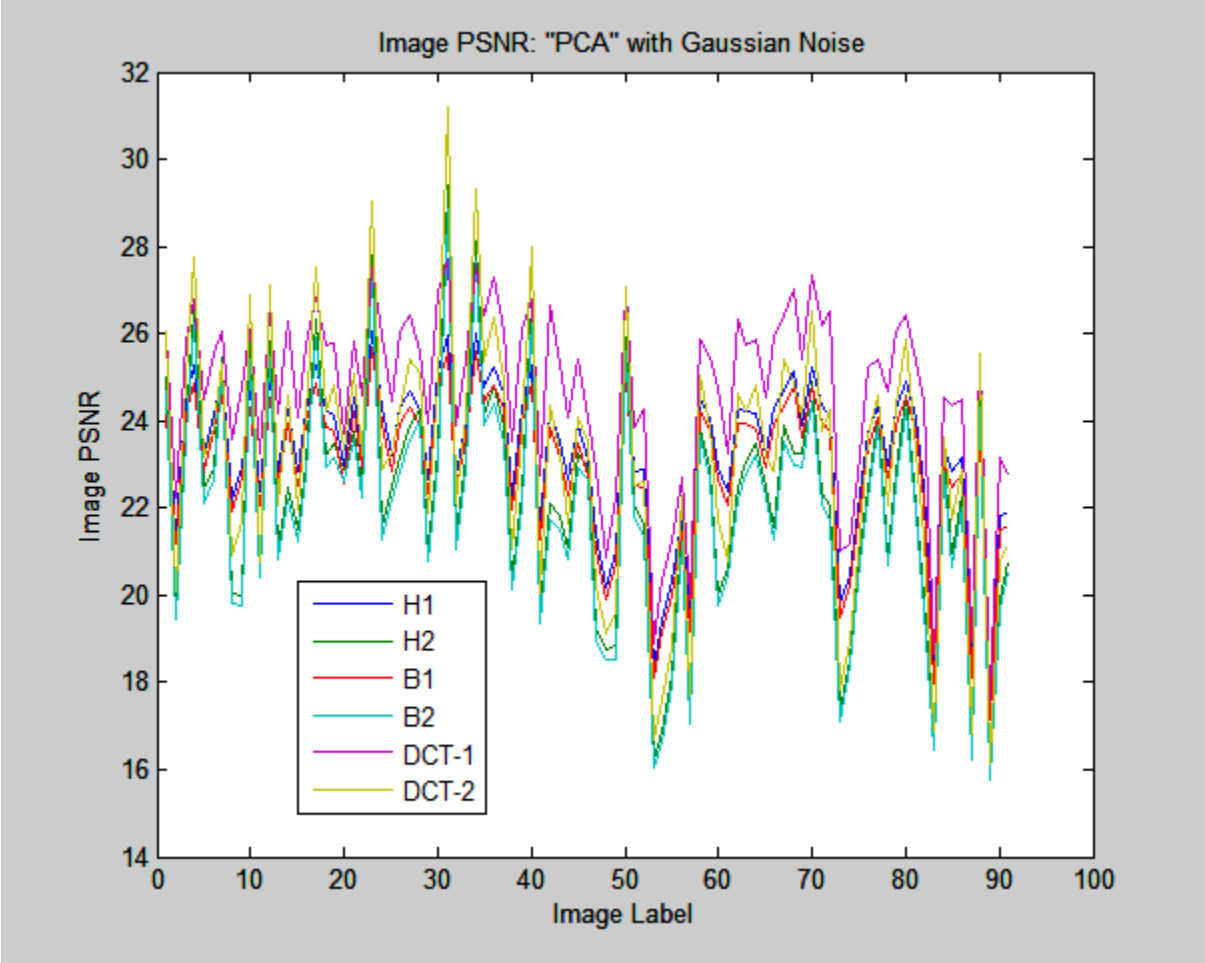

Figure 5.17 PSNR of Multi-resolution DCT and Wavelets under Gaussian Noise for 'PCA'

Figure 5.17 shows PSNR generated by our algorithm, Haar and Biorthogobal 1.5 wavelets under 'Gaussian' noise for the dataset 'PCA'. There is a purple line and a yellow line, representing the PSNR generated by our algorithm in level one and level two respectively. The purple line is usually at the top of the figure, with some cases that the yellow line goes above the purple line. There is a green line and light blue line, representing the PSNR generated by Haar and Biorthogobal 1.5 wavelets in level two respectively. The green line and light blue line are

usually below the yellow line. Therefore, in the same level, our algorithm always generates the highest PSNR, that is our algorithm has the highest computation accuracy in term of PSNR.

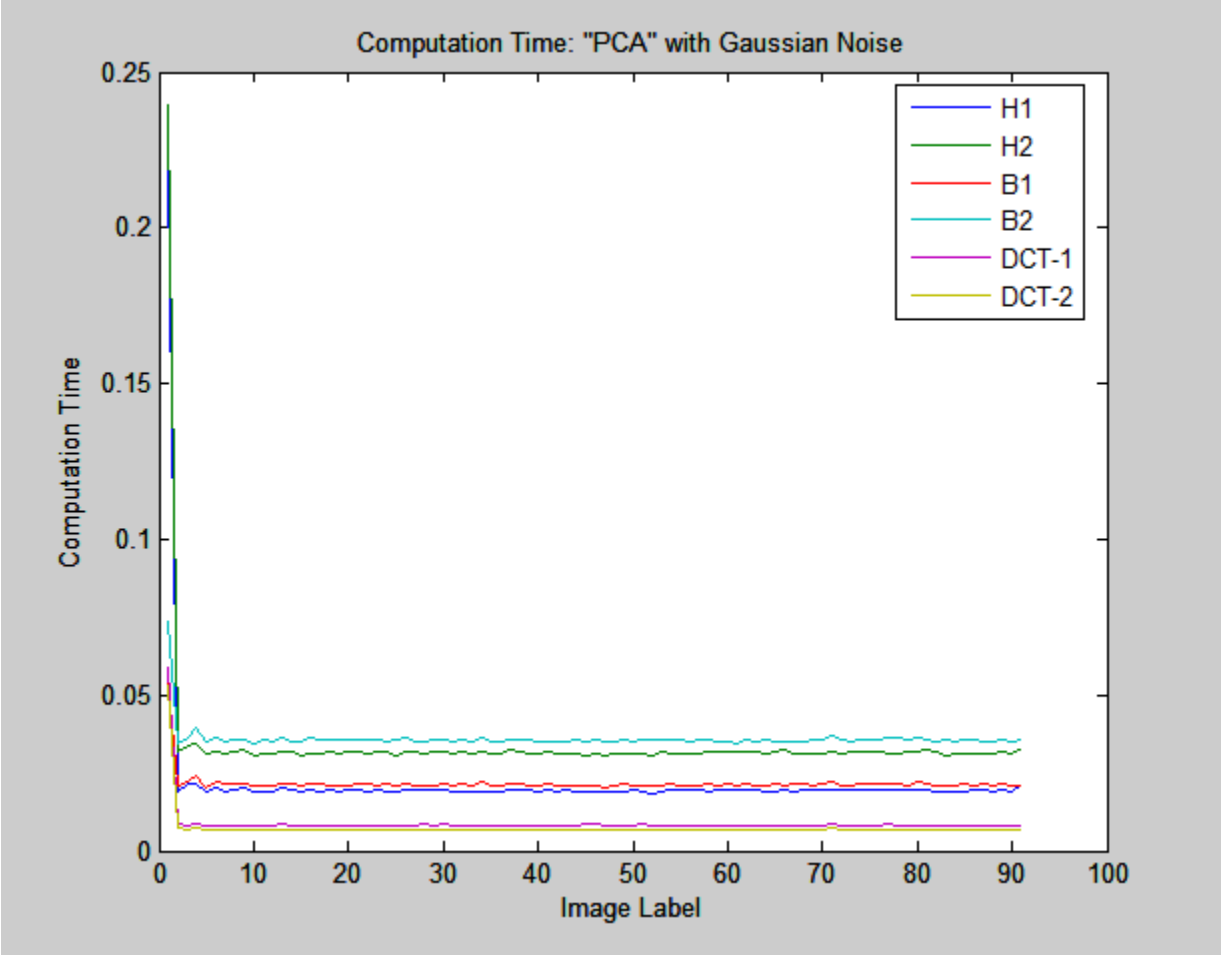

Figure 5.18 Computation Time of Multi-resolution DCT and Wavelets under Gaussian Noise for 'PCA'

Figure 5.18 shows computation time cost by our algorithm, Haar and Biorthogobal 1.5 wavelets under 'Gaussian' noise for the dataset 'PCA'. At the bottom of this figure, there is a yellow line and a purple line, representing the computation time cost by our algorithm in level two and one respectively. Therefore, our algorithm has the lowest computation time, that is our algorithm has the highest computation efficiency.

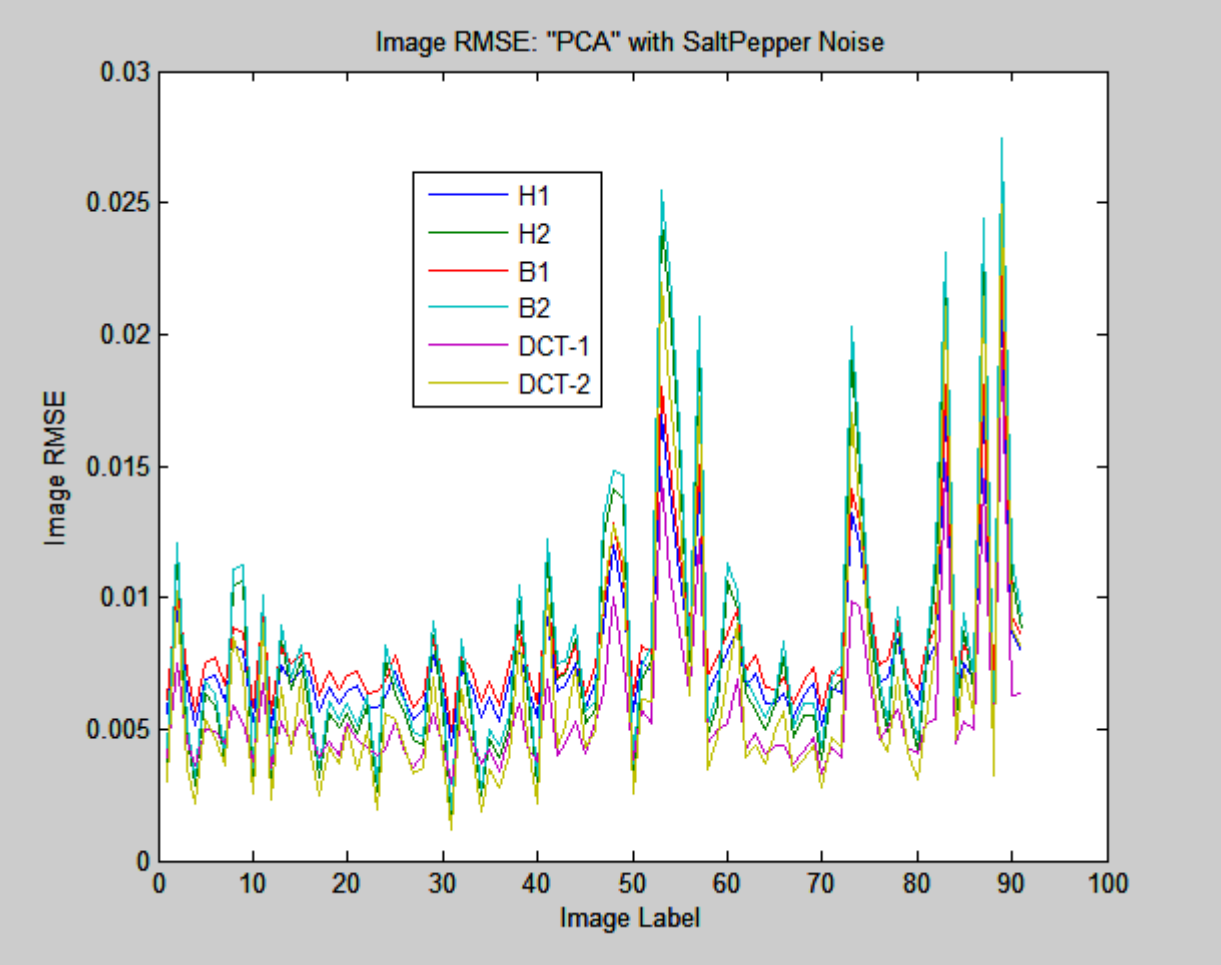

Figure 5.19 RMSE of Multi-resolution DCT and Wavelets under 'Salt & Pepper' Noise for 'PCA'

Figure 5.19 shows RMSE generated by our algorithm, Haar and Biorthogobal 1.5 wavelets under 'Salt & Pepper' noise for the dataset 'PCA'. There is a purple line and a yellow line, representing the RMSE generated by our algorithm in level one and level two respectively; these two lines are usually at the bottom of the figure. There is a green line and light blue line, representing the RMSE generated by Haar and Biorthogobal 1.5 wavelets in level two respectively; these two lines are usually above the yellow line and purple line. Therefore, in the same level, our algorithm always generates the lowest RMSE, that is our algorithm has the highest computation accuracy in term of RMSE.

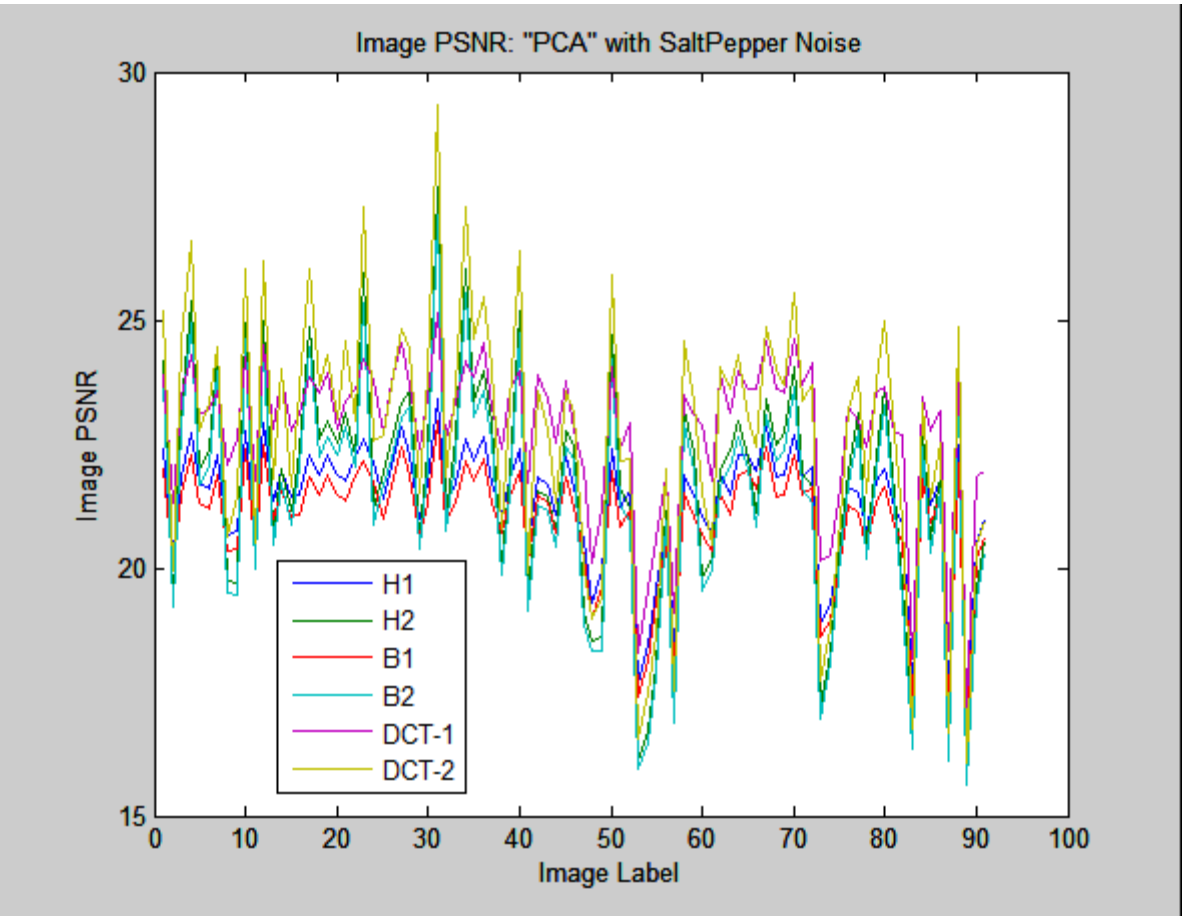

Figure 5.20 PSNR of Multi-resolution DCT and Wavelets under 'Salt & Pepper' Noise for 'PCA'

Figure 5.20 shows PSNR generated by our algorithm, Haar and Biorthogobal 1.5 wavelets under 'Salt & Pepper' noise for the dataset 'PCA'. There is a purple line and a yellow line, representing the PSNR generated by our algorithm in level one and level two respectively: these two lines are usually at the top of the figure. There is a green line and light blue line, representing the PSNR generated by Haar and Biorthogobal 1.5 wavelets in level two respectively; these two lines are usually below the yellow line and the purple line. Therefore, in the same level, our algorithm always generates the highest PSNR, that is our algorithm has the highest computation accuracy in terms of PSNR.

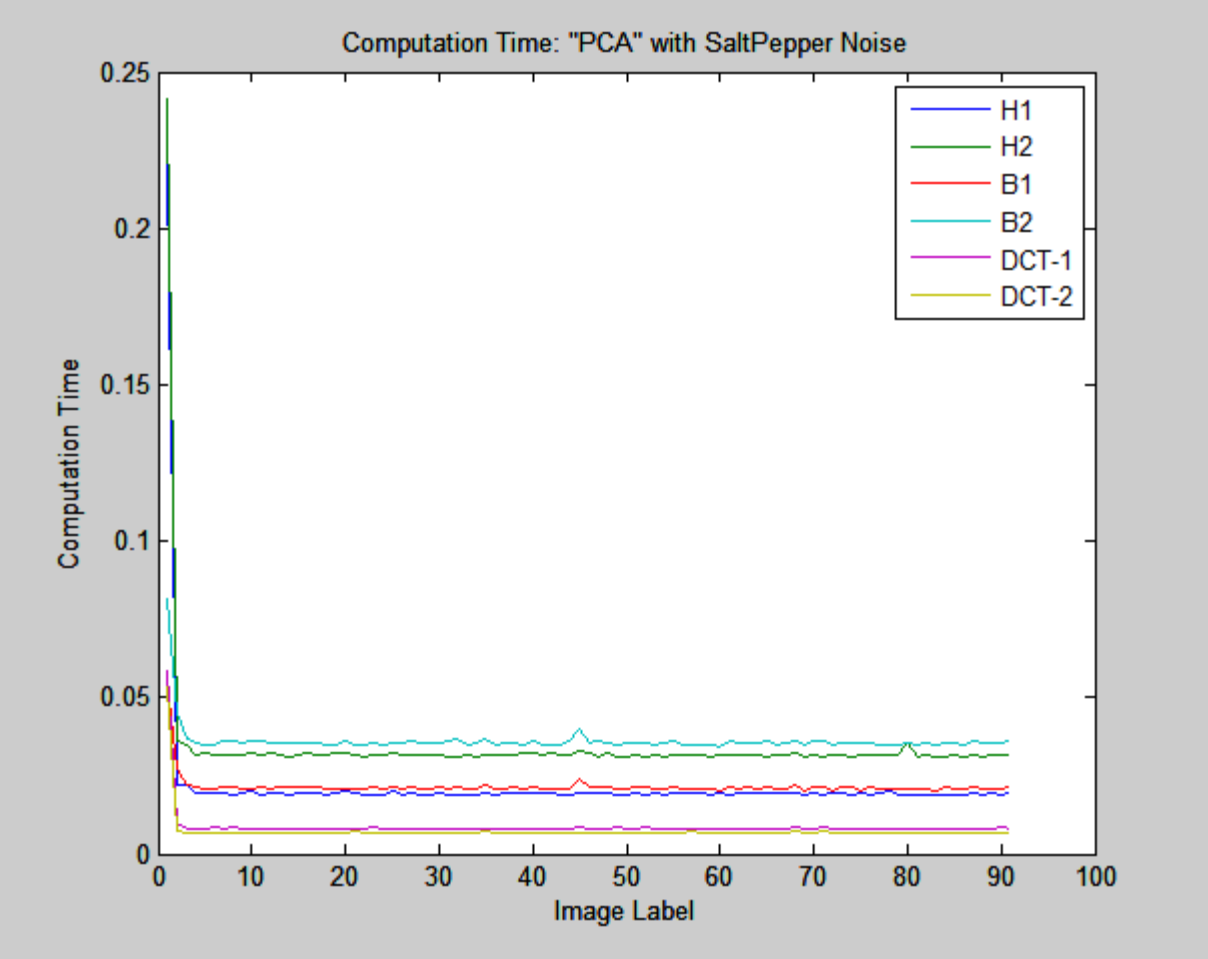

Figure 5. 21 Computation Time of Multi-resolution DCT and Wavelets under 'Salt & Pepper' Noise for 'PCA'

Figure 5.21 shows computation time cost by our algorithm, Haar and Biorthogobal 1.5 wavelets under 'Salt & Pepper' noise for the dataset 'PCA'. At the bottom of this figure, there is a yellow line and a purple line, representing the computation time cost by our algorithm in level two and level one respectively. Therefore, our algorithm has the lowest computation time, that is our algorithm has the highest computation efficiency.

The simulation clearly demonstrates that our algorithm outperforms wavelets in terms of computation accuracy and efficiency.

Summary

In this Chapter, we propose a novel image compression and denoising algorithm based on our multi-resolution scheme. Our algorithm has advanced computation accuracy and efficiency. It can be adaptive into different kinds of image descriptors. We take Fourier and multi-resolution discrete cosine transform (DCT) as two examples. Our algorithms based on multi-resolution Fourier and multi-resolution discrete cosine transform (DCT) both outperform wavelets in terms of computation accuracy and efficiency.

6 CONCLUSIONS

Image descriptors play an important role in image representation and analysis. Current image descriptors mainly face four challenges: high computation efficiency, low storage requirements, robustness to noise and good flexibility to represent an image in different views.

Multi-resolution image descriptors can represent and analysis an image in different resolution levels. The high resolutions provide detail information while the low resolutions show general views. This hierarchical structure identifies important features of an image among the unimportant ones; therefore we can efficiently compact images and meet the low storage requirements. Furthermore, the values of noise data are usually different with general image data. Through performing statistical computation of the coefficients from the same or different resolution levels, we can easily filter the noise and increase the robustness. And also, the nature of multi-resolution representation greatly improves the flexibility to represent an image.

Therefore, multi-resolution image descriptors are good candidates to deal with these challenges. However, current multi-resolution descriptors suffer different kinds of limitations. Wavelet descriptors have been widely used in multi-resolution image analysis. However, making the wavelet transform shift and rotation invariant produces redundancy and requires complex matching processes. As to other multi-resolution descriptors, such as image pyramid, scale space method, fractal geometry, binary partition trees, adaptive grid resolution; they usually depend on other methods, such as filtering function, prior-domain knowledge, etc.; that not only increases the computation complexity, but also changes the original image information.

We propose a novel odd-even multi-resolution image representation and analysis scheme. This scheme can analyze images recursively and effectively without introducing artifacts or

changes to the original image, provide multi-resolution images, obtain higher resolution images only using information from lower resolution levels and be implemented in parallel processing. This scheme has the high computation efficiency, good invariance properties, strong capability to the noise, and good flexibility to represent and analyze image in different ways. This scheme can be applied in many image processing areas, such as image compression, noise, filtering feature extraction.

Our multi-resolution scheme is based on transforming each image into an odd-even image tree. Through applying image descriptors to this odd-even image tree, we get the relative multi-resolution image descriptors. Multi-resolution analysis is based on downsampling expansion with maximum energy extraction followed by upsampling reconstruction. Since the maximum energy retained in the lowest frequency coefficients; we do maximum energy extraction through keeping the P lowest coefficients from each resolution level.

Since there are mainly two kinds of image descriptors: boundary-based and region-based descriptors; our multi-resolution scheme also has relative two formats: boundary-based scheme which generates boundary-based multi-resolution descriptors and region-based scheme which generates region-based multi-resolution descriptors.

We apply our multi-resolution scheme in the areas of feature extraction, image compression and denoising.

Feature extraction plays important roles in image processing and pattern recognition. Because analysis with a large number of variables generally requires a large amount of memory and computation power or makes a classification algorithm over fit the training sample and

generalize poorly to new samples. However, the current feature extraction algorithms are either restricted by assumptions or hard to be implemented or suffering high computation complexity.

Exploring features in different resolutions reveals the dominant coefficients (features) in comparison to the redundant ones. We propose a novel feature extraction algorithm based on our multi-resolution scheme. Our algorithm is free from assumption, robust to noise, adaptive to any kind of image descriptor, and can be implemented by parallel processing.

In special, we take Cartesian Zernike moments as an example. Since it is difficult to define the most effective Zernike coefficients in a concise way; the most popular method is to use all coefficients under Zernike order n as image features. This method increases computation complexity and may generate computational errors due to the redundant features. An alternate method is to use the first P coefficients under the given Zernike order n. This method arbitrarily ignores the importance of other coefficients and is not optimized. Our algorithm based on multi-resolution Cartesian Zernike moments extracts effective coefficients with stronger ability to reveal essential characteristics of an image and distinguish this image with other images.

The purpose of image compression is to reduce irrelevance and redundancy of the image data for storing or transmitting it efficiently with preserving its quality for the given application. Since noise is usually unavoidable in the real world image data. If we can combine image compression with image denoising, that is compacting an image through reducing its noise, it will be very prospective.

Currently, the most popular image descriptor that can do image compression and denoising at the same time is wavelets; however wavelets suffer high computation complexity and poor invariance property.

We propose a novel image compression and denoising algorithm based on our multi-resolution scheme. Our algorithm has outstanding performance in compressing data and filtering noise; it is free from block artifacts, adaptive to any kinds of image descriptors and ideal for parallel computing.

In special, we take Fourier and discrete cosine transform (DCT) as two examples. Although traditional Fourier transform has outstanding computation efficiency and excellent invariance property, it usually cannot outperform wavelets in the area of image compression and denoising, due to lacking multi-resolution property. Traditional DCT has strong energy compaction property; however, its performance generally degrades at high compression ratios mainly due to the underlying block-based strategy; therefore we usually consider wavelets are more advanced than DCT in the area of image compression. We propose two image compression and denoising algorithms; one is based on multi-resolution Fourier; another is based on multi-resolution DCT; both these algorithms outperform wavelets in terms of computation accuracy and efficiency.

In the future, we plan to extend our research to these three points

- Generate more multi-resolution image descriptors to deal with more real world problems, such as image segmentation, information retrieval.
- Extend our multi-resolution scheme for processing 3D images.
- Combine our multi-resolution image scheme with text and video processing for multi-media processing.

REFERENCES

[1] R. C. Veltkamp and M. Hagedoorn. Principles of Visual Information Retrieval. London, U.K.: Springer-Verlag, 2000.

[2] Hanchuan Peng. "Bioimage informatics: a new area of engineering biology". Bioinformatics, Vol. 24, No. 17, pp. 1827-1836, 2008.

[3] A. Pentland, R. W. Picard, and S. Sclaroff. "Photobook: Content-based manipulation of image databases". International Journal of Computer. Vision. vol. 18, no. 3, pp. 233–254, Jun. 1996.

[4] Ioannis Pitas and Constantine Kotropoulos. Nonlinear Model-Based Image/Video Processing and Analysis. John Wiley & Sons, Inc., New York, NY, 2001.

[5] C.-Y. Wee, P. Raveendran, and F. Takeda, "New computational methods for full and subset Zernike moments," Information Sciences 159(2004), pp. 203-220.

[6] Novotni, M.; Klein, R. (2004). "Shape retrieval using 3D Zernike descriptors". Computer Aided Design 36 (11): 1047–1062.

[7] Farokhi, Sajad; Shamsuddin, Siti Mariyam; Sheikh, U.U; Flusser, Jan (2014). "Near Infrared Face Recognition: A Comparison of Moment-Based Approaches". The 8th International Conference on Robotic, Vision, Signal Processing & Power Applications. 291 (1). Springer. pp. 129–135.

[8] Gu, J.; Shu, H. Z.; Toumoulin, C.; Luo, L. M. (2002). "A novel algorithm for fast computation of Zernike moments". Pattern Recogn. 35 (12): 2905–2911.

[9] Lundström, L.; Unsbo, P. (2007). "Transformation of Zernike coefficients: scaled, translated and rotated wavefronts with circular and elliptical pupils". J. Opt. Soc. Am. A 24 (3): 569-577.

[10] G. Papari, P. Campisi, N. Petkov and A. Neri. ” A biologically motivated multiresolution approach to contour detection”, EURASIP Journal on Advances in Signal Processing, Vol. 2007, Article ID 71828, 28 pages, 2007.

[11] T. Zahn and R. Z. Roskies. "Fourier descriptors for plane closed curves". IEEE Transactions on Computers, vol. 21, pp. 269--281, 1972.

[12] Xuwei Liang, Qi Zhuang, Ning Cao and Jun Zhang. "Shape modeling and clustering of white matter fiber tracts using Fourier descriptors”. IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), 2009. Nashville, TN, pp. 292-297.

[13] K. M. Hosny, “A systematic method for efficient computation of full and subsets Zernike moments,” Information Sciences 180 (2011), pp. 2299-2313.

[14] Arrate Muñoz, Raphaël Ertlé and Michael Unser(2002). “Continuous wavelet transform with arbitrary scales and $O(N)$ complexity”. Signal Processing 82(5): 749-757.

[15] Y. Andreopoulos, A. Munteanu, G. Van der Auwera, J. Cornelis and P. Schelkens. "Complete-to-overcomplete discrete wavelet transforms: theory and applications". IEEE Transaction on Signal Processing, vol. 53, no. 4, pp. 1398-1412, April 2005.

[16] S. Junding and W. Xiaosheng. "Chain Code Distribution-Based Image Retrieval". International Conference on Intelligent Information Hiding and Multimedia Signal Processing, China, 2006, pp. 139 -142.

[17] F. Mokhtarian and A. Mackworth. "Scale-based description and recognition of planar curves and two-dimensional shapes". IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, pp. 34-43, 1986.

[18] Chazelle, Bernard (1993), "An optimal convex hull algorithm in any fixed dimension", Discrete and Computational Geometry 10 (1): 377–409.

[19] D. S. Zhang and G. Lu. "Study and Evaluation of Different Fourier Methods for Image Retrieval". International Journal of Computer Vision, vol. 23, pp. 33-49, 2005.

[20] H. Kauppinen, T. Seppänen, and M. Pietikäinen. "An Experimental Comparison of Autoregressive and Fourier-Based Descriptors in 2D Shape Classification". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 17, No. 2, 1995, pp. 201-207.

[21] B.M. Mehtre, M.S. Kankanhalli, W.F. Lee. "Shape Measures for Content Based Image Retrieval: A Comparison". Information Processing Management, Vol. 33, No 3, pp.319-337, 1997.

[22] D. Zhang and G. Lu. "A Comparative Study of Curvature Scale Space and Fourier Descriptors for Shape- Based image Retrieval". Journal of Visual Communication and Image Representation, Vol. 14, No. 1, 2003, pp. 41-60.

[23] Costa, L.F. and Cesar, R.M., Shape Analysis and Classification, Theory and Practice. CRC Press, Boca Raton, Florida. 2001.

[24] C.-W. Chong, P. Raveendran, and R. Mukundan, "Translation Invariants of Zernike Moments", Pattern Recognition, 36 (2003), pp. 1765-1773.

[25] E. Persoon and K. Fu, “Shape Discrimination Using Fourier Descriptors,” IEEE Transactions on Systems, Man, and Cybernetics, Vol. 7, pp. 170-179, 1997.

[26] Y. Mei and D. Androutsos, "Robust Affine Invariant Region-based Shape Descriptors: the ICA Zernike Moment Shape Descriptor and the Whitening Zernike Moment Shape Descriptor", IEEE Signal Processing Letters, Vol. 16, No. 10, pp. 877-880, October 2009.

[27] Hokyung Kang, Yong Man Ro and Sung Min Kim. “A Microcalcification Detection Using Adaptive Contrast Enhancement on Wavelet Transform and Neural Network". IEICE Transaction. on Information & Systems ,Vol.E89-D,NO.3 pp.1280 - 1287 March 2006.

[28] G.K. Rohde, A.J.S. Ribeiro, K.N. Dahl and R.F. Murphy. “Deformation-based nuclear morphometry: Capturing nuclear shape variation in HeLa cells”. Cytometry, vol 73A, pp. 341-350, 2008.

[29] S. Sanei and J. Chambers. EEG Signal Processing. John Wiley & Sons, 2007.

[30] Athanasios Mademlis, Petros Daras, Dimitrios Tzovaras, Michael G. Strintzis. “3D object retrieval using the 3D shape impact descriptor”. Pattern Recognition 42(11): 2447-2459, 2009

[31] Apostolos Axenopoulos, Petros Daras, Dimitrios Zarpalas, Dimitrios Tzovaras and Michael G. Strintzis. “3D content-based search and retrieval using the 2D polar wavelet transform”.IEEE International Conference on Image Processing Genoa, Italy Sep.11-14 2005

[32] P. Campisi, A. Neri, M. Visconti, "A wavelet based method for high frequency subbands watermark embedding", SPIE Multimedia Systems and Applications III, 5-8 November 2000, Boston, USA.

[33] I. Kunttua, L. Lepistöa, J. Rauhamaab,and A. Visaa, “Multiscale Fourier descriptors for defect image retrieval, ” Pattern Recogn. Lett., vol. 27, no. 2, pp. 123–132, Jan. 2006.

[34] V. Vilaplana, F. Marques and P. Salembier. “Binary Partition Trees for Object Detection”. IEEE Transactions on Image Processing, Volume 17, Issue 11, Page(s):1 - 16, Nov. 2008.

[35] Rafael C. Gonzalez and Richard E.Woods. Digital Image Processing (the third edition). Pearson Prentice Hall 2008.

[36] P. Salembier, L. Garrido. “Binary Partition Tree as an Efficient Representation for Image Processing, Segmentation, and Information Retrieval” .IEEE Transactions on Image Processing, 9(4):561-576, April 2000.

[37] G. Papari, P. Campisi, A. Neri, N. Petkov, “Contour detection by multiresolution surround inhibition”, IEEE International Conference on Image Processing (ICIP 2006), October 2006, Atlanta, USA, pp:749-752.

[38] Mingqiang Yang, Kidiyo Kpalma, Joseph Ronsin. A Survey of Shape Feature Extraction Techniques. Peng-Yeng Yin. Pattern Recognition, IN-TECH, pp.43-90, 2008. <hal-00446037> .

[39] K. Chakrabarti, M. Binderberger, K. Porkaew, and S. Mehrotra, “ Similar shape retrieval in mars. “ in Proc. IEEE International Conference on Multimedia and Expo, 2000, Vol .2 pp. 709-712.

[40] K. kpalma and J. Ronsin, “Multiscale contour description for pattern recognition”. Pattern Recognition Letters, vol. 27, pp. 1545-1559, 2006.

[41] Bruno Garguet-Duport, Jacky Girel, Jean-Marc Chassery, and Guy Pautou (1996) “The use of multiresolution analysis and wavelets transform for merging SPOT panchromatic and multispectral image data". Photogrammetric Engineering and remote sensing 62(9): 1057-1066.

[42] Larson, David R. (2007). "Unitary systems and wavelet sets". Wavelet Analysis and Applications. Appl. Numer. Harmon. Anal.. Birkhäuser. pp. 143–171.

[43] Barnsley, Michael (2000). Fractals everywhere. Second edition, Morgan Kaufmann.

[44] Hsu, T., Hu KJ. “Multi-resolution Texture Segmentation Using Fractal Dimension”. 2008 International Conference on Computer Science and Software Engineering, pp. 201–204.Wuhan China.

[45] Aiazzi, B., Alparone, L., Baronti, S., Garzelli,A. “Multi-resolution estimation of fractal dimension from noisy images”. J. Electron. Imaging, 10(1), pp. 339-348, 2001.

[46] Bai-ling Zhang; Wenjin Lu.“Multiresolution fractal analysis and classification of neurite images”. Biomedical Engineering and Informatics (BMEI), 2010 3rd International Conference on Volume: 1 , Page(s): 419 – 423

[47] A. Jacquin, “Image coding based on a fractal theory of iterated contractive image transformations”. IEEE Trans. Image Processing, vol. 1, pp. 18-30, 1992.

[48] B. Chinna Rao, Dr. M. Madhavi Latha “Analysis of Multi Resolution Image Denoising Scheme using Fractal Transform”. The International journal of Multimedia & Its Applications (IJMA) Vol.2, No.3, pp. 63-74, August 2010.

[49] Dengsheng Zhang, Guojun Lu. “Review of shape representation and description techniques”. Pattern Recognition, Vol. 37, No. 1. (2004), pp. 1-19

[50] I. Yong, J. Walker, J. Bowie, “An analysis technique for biological shape”, Comput. Graphics Image Process. 25 (1974): 357–370.

[51] M. Peura, J. Iivarinen, “Efficiency of simple shape descriptors”. Proceedings of the Third International Workshop on Visual Form, Capri, Italy, May, 1997, pp. 443–451.

[52] D.P. Huttenlocher, W.J. Rucklidge, “A multi-resolution technique for comparing images using the Hausdorff distance”. Technical Report, TR-92-1321, Department of Computer Science, Cornell University, 1992.

[53] W.J. Rucklidge, “Efficient locating objects using Hausdorff distance”, Int. J. Comput. Vision 24 (3) (1997): 251–270.

[54] S. Belongie, J. Malik, J. Puzicha, “Matching shapes”. Proceedings of Eighth IEEE International Conference on Computer Vision (ICCV2001), Vol. I, Vancouver, Canada, July, 2001, pp. 454–461.

[55] E.R. Davies. Machine Vision: Theory, Algorithms, Practicalities. Academic Press Academic Press; 4 edition, 2012.

[56] P.J. van Otterloo, “A Contour-Oriented Approach to Shape Analysis”, Prentice-Hall International (UK) Ltd, Englewood Cliffs, NJ, 1991, pp. 90–108.

[57] D.S. Zhang, G. Lu, “A comparative study of Fourier descriptors for shape representation and retrieval”. Proceedings of the Fifth Asian Conference on Computer Vision (ACCV02), Melbourne, Australia, January 22–25, 2002, pp. 646–651.

[58] M. Sonka, V. Hlavac, R. Boyle. Image Processing, Analysis and Machine Vision. Cengage Learning; 3 edition 2007.

[59] K.L. Kashyap, R. Chellappa, “Stochastic models for closed boundary analysis: representation and reconstruction”, IEEE Trans. Inform. Theory 27 (1981): 627–637.

[60] Y. He, A. Kundu, “2D shape classiffication using hidden Markov model”, IEEE Trans. Pattern Anal. Mach. Intell. 13 (11) (1991): 1172–1184.

[61] I. Sekita, T. Kurita, N. Otsu, “Complex autoregressive model for shape recognition”, IEEE Trans. Pattern Anal. Mach. Intell. 14 (1992): 489–496.

[62] H. Asada, M. Brady. The curvature primal sketch. Ft. Belvoir Defense Technical Information Center FEB 1984.

[63] H. Asada, M. Brandy, “The curvature primal sketch”, IEEE Trans. Pattern Anal. Mach. Intell. 8 (1) (1986) pp. 2-14

[64] J. W. McKee and J. K. Aggarwal. "Computer recognition of partial views of curved objects", IEEE Trans. Comput., vol. C-26, no. 8, pp.790 -800, 1977.

[65] M. Daoudi, S. Matusiak, “Visual image retrieval by multiscale description of user sketches”, J. Visual Lang. Comput. 11(2000): 287–301.

[66] D.H. Eberly, Geometric methods for analysis of ridges in N-dimensional images, Ph.D. Thesis, University of North Carolina at Chapel Hill, 1994.

[67] J.R. Ohm, F.B. Bunjamin, W. Liebsch, B. Makai, K. Muller, A. Somlic, D. Zier, “A set of visual feature descriptors and their combination in a low-level description scheme”, Signal Process. Image Commun. 16 (2000): 157–179.

[68] Q.M. Tieng, W.W. Boles, “Recognition of 2D object contours using the wavelet transform zero-crossing representation”, IEEE Trans. Pattern Anal. Mach. Intell. 19 (8) (1997): 910–916.

[69] H.S. Yang, S.U. Lee, K.M. Lee, “Recognition of 2D object contours using starting-point-independent wavelet coefficient matching”, J. Visual Commun. Image Represent. 9 (2) (1998): 171–181.

[70] R. Chellappa, R. Bagdazian, “Fourier coding of image boundaries”, IEEE Trans. Pattern Anal. Mach. Intell. 6 (1) (1984): 102–105.

[71] H. Freeman, “On the encoding of arbitrary geometric configurations”, IRE Trans. Electron. Comput. EC-10 (1961): 260–268.

[72] W.I. Groskey, R. Mehrotra, “Index-based object recognition in pictorial data management”, Comput. Vision Graphics Image Process. 52 (1990): 416–436.

[73] W.I. Groskey, P. Neo, R. Mehrotra, “A pictorial index mechanism for model-based matching”, Data Knowledge Eng. 8 (1992): 309–327.

[74] S. Berretti, A.D. Bimbo, P. Pala, “Retrieval by shape similarity with perceptual distance and effective indexing”, IEEE Trans. Multimedia 2 (4) (2000): 225–239.

[75] G. Dudek, J.K. Tsotsos. “Shape representation and recognition from multiscale curvature, Computing”. Vision Image Understanding 68 (2) (1997): 170–189.

[76] R.L. Kennedy, Y. Lee, B.V. Roy, C.D. Reed, R.P. Lippmann. Solving Data Mining Problems Through Pattern Recognition. Prentice-Hall, PTR, Upper Saddle River, NJ, 1998, pp. 11–16 (Chapter 9).

[77] B. Jahne. Digital Image Processing—Concepts, Algorithms and Scientific Applications. Springer, Berlin, Heidelberg, Springer; 3rd edition, 1995

[78] M.K. Hu, "Visual pattern recognition by moment invariants", IRE Trans. Inf. Theory IT-8 (1962) Volume 8, Issue: 2 pp. 179–187.

[79] D.S. Zhang, Image retrieval based on shape, Ph.D. Thesis, Monash University, Australia, March, 2002.

[80] G. Taubin, D.B. Cooper, "Recognition and positioning of rigid objects using algebraic moment invariants", SPIE Conference on Geometric Methods in Computer Vision, Vol. 1570, University of Florida, Florida, USA 1991, pp. 175–186.

[81] G. Taubin, D.B. Cooper, "Object recognition based on moment (or algebraic) invariants", Geometric Invariance in Computer Vision, MIT Press, Cambridge, MA, 1992, pp. 375–397.

[82] B. Scassellati, S. Slexopoulos, M. Flickner (1994), "Retrieving images by 2D shape: a comparison of computation methods with human perceptual judgments", in: SPIE Proceedings on Storage and Retrieval for Image and Vid eo Databases II, Vol. 2185, pp. 2–14.

[83] C.-H. Teh, R.T. Chin, "On image analysis by the methods of moments", IEEE Trans. Pattern Anal. Mach. Intell. 10 (4) (1988): 496–513.

[84] Y.-S. Kim, W.-Y. Kim, "Content-based trademark retrieval system using a visually salient feature", Image Vision Comput. 16 (1998): 931–939.

[85] W.-Y. Kim, Y.-S. Kim, “A region-based shape descriptor using Zernike moments”, Signal Process. Image Commun. 16 (2000): 95–102

[86] M.R. Teague(1980), “Image analysis via the general theory of moments”, J. Opt. Soc. Am. 70 (8): 920–930.

[87] D.S. Zhang, G. Lu, “Enhanced generic Fourier descriptor for object-based image retrieval”, Proceedings of the IEEE International Conf. Acoustics, Speech, and Signal Processing (ICASSP2002), Vol. 4, Orlando, FL, USA, May 13–17, 2002, pp. 3668–3671.

[88] G.J. Lu, A. Sajjanhar, “Region-based shape representation and similarity measure suitable for content-based image retrieval”, Multimedia Syst. 7 (2) (1999) 165–174.

[89] King-Chu Hung, Chih-Liang Chen, Jyh-Ming Kuo “The generalized uniqueness wavelet descriptor”. Proceedings of International Conference Image Processing (ICIP) Kobe, Oct. 1999 vol.1 pp.600-604.

[90] Chui, Charles K. (1992). An Introduction to Wavelets. San Diego: Academic Press.

[91] Bingsheng Wu and Chaozhi Cai. “Wavelet Denoising and Its Implementation in LabVIEW”. The 2nd International Congress on Image and Signal Processing (CISP '09), Oct. 2009 pp. 1-4.

[92] Chuang, G.C.-H. and Kuo, C.-C.J. “Wavelet descriptor of planar curves: theory and applications”. IEEE Transaction on Image Processing vol.5 pp.56-70.

[93] Yujing Zhang, Qirui Han “Edge Detection Algorithm Based on Wavelet Transform and Mathematical Morphology”. International Conference on Control, Automation and Systems Engineering (CASE),July 2011, pp.1-3.

[94] M.Ceylan, O.N.Ucan, Y.Özbay, R.Jennane, G.Aufort, C.L.Benhamou, "Comparison of discrete wavelet transform and complex wavelet transform in hybrid skeletonization based on cvann ", İstanbul Aydın Üniversitesi, Fen Bilimleri Dergisi, 1, , 27-51.

[95] A. V. Oppenheim and J. S. Lim (1981). "The importance of phase in signals" proceedings of IEEE, Vol. 69, pp. 529-541.

[96] Edward H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden(1984) "Pyramid Method in Image Processing", RCA Engineer, vol. 29(6), pp. 33-41.

[97] Dube, S.; Hong, L. "Adaptive image pyramid based compression algorithm" Conference Record of the Thirty-Fourth Asilomar Conference on Signals, Systems and Computers, San Diego, CA, Oct.-Nov. 2000. Vol.2, pp.1045-1048.

[98] Pauwels, E., van Gool, L., Fiddelaers, P. and Moons, T.: "An extended class of scale-invariant and recursive scale space filters", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 17, No. 7, pp. 691–701, 1995.

[99] Lindeberg, Tony. "Scale-space theory: A basic tool for analysing structures at different scales". Journal of applied statistics. 21(2): 224-270 1994.

[100] T. Lindeberg (2014). Scale selection, Computer Vision: A Reference Guide, (K. Ikeuchi, Editor), Springer.

[101] Witkin, A. P. "Scale-space filtering", Proc. 8th Int. Joint Conf. Art. Intell., Karlsruhe, Germany,1019--1022, 1983.

[102] Carlos W.D. de Almeida, Renata M.C.R. de Souza, Carlos E.B. Rodrigues and Nicomedes L. Cavalcanti Junior. "Image retrieval using the curvature scale space (CSS)

technique and the self-organizing map (SOM) model under rotation" the 2nd International Conference on Digital Information Management (ICDIM '07). Lyon, Oct. 2007, Vol. 01 pp. 357 – 361.

[103] R. Lopes and N. Betrouni. "Fractal and multifractal analysis: A review" Medical Image Analysis vol. 13, pp. 634–649, 2009.

[104] Liu, Jing Z.; Zhang, Lu D.; Yue, Guang H. (2003). "Fractal Dimension in Human Cerebellum Measured by Magnetic Resonance Imaging". Biophysical Journal 85 (6): 4041–4046.

[105] S. Belkasim, E. Hassan and T.Obeidi "Explicit Invariance of Cartesian Zernike Moments" Pattern Recognition Letters, 28(2007), pp. 1969–1980.

[106] Honarvar Shakibaei Asli, Barmak; Raveendran, Paramesran (July 2013). "Recursive formula to compute Zernike radial polynomials" Opt. Lett. (OSA) 38 (14): 2487–2489.

[107] S. X. Liao, Image Analysis by Moments, Ph.D. dissertation, The University of Manitoba, 1993.

[108] M. Pawlak: "On the reconstruction aspect of moment descriptors", IEEE Transaction Information & Theory, 38(1992), pp. 1698–1708.

[109] Yanjun Zhao and Saeid Belkasim. "Multi-resolution Fourier Descriptors for Multi-resolution Shape Analysis" IEEE Signal Processing Letters vol. 10, no. 19, pp: 692-695 Oct. 2012.

[110] Yanjun Zhao and Saeid Belkasim. "Feature Extraction based on Multi-resolution Cartesian Zernike for Biomedical Image processing". 9th International Symposium on

Bioinformatics Research and Applications (ISBRA) Charlotte, North Carolina, May 20-22, 2013.

[111] N. Ahmed, T. Natarajan and K.R.Rao. (1974 Jan.). “Discrete Cosine Transform". IEEE Trans. Computers, C (32), pp. 90-93.

[112] Tahmasbi, A.; Saki, F.; Shokouhi, S.B.. (2010). "An Effective Breast Mass Diagnosis System using Zernike Moments". 17th Iranian Conf. on Biomedical Engineering (ICBME'2010). Isfahan, pp. 1–4.

[113] Arai, Y.; Agui, T.; Nakajima, M. (November 1988). "A fast DCT-SQ scheme for images". IEICE Transactions 71 (11): 1095–1097.

[114] Triantafyllidis, G.A.; Tzovaras, D.; Strintzis, M.G. “Blocking artifact detection and reduction in compressed data”. IEEE Transactions on Circuits and Systems for Video Technology, Volume:12 , Issue: 10 pp: 877 – 890.

[115] The cancer imaging archive, National Cancer Institute https://www.cancerimagingarchive.net/.

[116] Jolliffe I.T. “Principal Component Analysis”. Series: Springer Series in Statistics, 2nd ed., Springer, NY, 2002, XXIX, 487 p. 28 illus. ISBN 978-0-387-95442-4.

[117] M. Aizerman, E. Braverman, and L. Rozonoer (1964). "Theoretical foundations of the potential function method in pattern recognition learning". Automation and Remote Control 25: 821–837.

[118] Heiko Hoffmann. “Kernel PCA for novelty detection”. Pattern Recognition. Volume 40, Issue 3, March 2007, pp. 863–874.

[119] M. Bernstein, V. De Silva, J. C. Langford, and J. B. Tenenbaum, "Graph approximations to geodesics on embedded manifolds," Technical report, Department of Psychology, Stanford University, Tech. Rep., 2000.

[120] J. B. Tenenbaum, V. de Silva, J. C. Langford. "A Global Geometric Framework for Nonlinear Dimensionality Reduction". Science 290, (2000), 2319–2323.

[121] S. T. Roweis and L. K. Saul. "Nonlinear Dimensionality Reduction by Locally Linear Embedding". Science 290, (2000), 2323–2326.

[122] D. Donoho and C. Grimes, "Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data". Proc Natl Acad Sci U S A. 2003 May 13; 100(10): 5591–5596.

[123] Ma, L.; Crawford, M. M.; Tian, J. W. (2010). "Generalised supervised local tangent space alignment for hyperspectral image classification". Electronics Letters 46 (7): 497-498.

[124] K. Q. Weinberger and L. K. Saul (2004). "Unsupervised learning of image manifolds by semidefinite programming". In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR-04), Washington D.C.

[125] Y. Bengio. "Learning Deep Architectures for AI". Foundations and Trends in Machine Learning. Jan. 1009 Vol. 2 Issue 1, pp.1-127.

[126] Koon-Pong Wong. "Fractal Image Coding for Emission Tomographic Image Compression" IEEE Nuclear Science Symposium Conference Record, Nov. 2001, vol 3 pp. 1376- 1379.

[127] Miss. S.S. Tamboli and Dr. V. R. Udupi "Image compression using the Haar wavelet transform". International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 8, Pages: 3166-3170, August 2013.

[128] A. K. Jain, "Image data compression: A review," Proc. IEEE, vol. 69, pp. 349–389, 1981.

[129] J. A. Storer, Image and Text Compression, Kluwer Academic Publisher, Norwell, MA, 1992.

[130] M. Nelson, Jean-Loup Gailly. The Data Compression Book, 2nd edition. Wiley, 1995.

[131] H. R. Wu, A.R. Reibman, W. Lin, F. Pereira and S.S. Hemami, "Perceptual visual signal compression and transmission", Proceedings of the IEEE, 2013, vol 101, pp. 2025 – 2043.

[132] R. Chalasani, J. C. Principe and N. Ramakrishnan. "A Fast Proximal Method for Convolutional Sparse Coding". The IEEE International Joint Conference on Neural Networks (IJCNN), Dallas, TX, Aug. 2013 http://cnel.ufl.edu/~rakesh/IJCNN_2013.pdf

[133] V. Velisavljevic, B. Beferull-Lozano and M. Vetterli, (2007 Jul.) "Sparse-frequency quantization for image compression with directionlets". IEEE Trans. Image Proc., 16(7), pp. 1761-1773.

[134] G. Piella and B. Pesquet-Popescu, "A Three-step Nonlinear Lifting Scheme for Lossless Image Compression", IEEE Inter. Conf. Signal Proc. (ICSP), San Antonio, TX, Sep. 16-Oct.19 2007. pp. 453-456.

[135] Y. Wang, M. T. Orchard, A. Vinay, V. Vaishampayan and A. R. Reibman. (2001 Mar.). "Multiple description coding using pairwise correlating transforms". IEEE Trans. Image Proc.,10(3) pp. 351-366.

[136] J. Zepeda, C. Guillemot and E. Kijak. (2011 Sep.) “Image Compression using the Iteration-tuned and Aligned Dictionary”. IEEE Jour. of Selected Topics in Signal Proc., special issue on Adaptive sparse representation of data and applications in signal and image proc., 5(5), pp. 1061-1073.

[137] S. Rane, P. Boufounos, A. Vetro and Y. Okada, "Low-Complexity Efficient Raw SAR Data Compression", SPIE Defense, Security and Sensing, Algorithms for Synthetic Aperture Radar Imagery, Orlando, FL, April 2011.

[138] L. J. Karam. Lossless Image Compression. The Essential Guide to Image Processing, Al Bovik Editor, Elsevier Academic Press, 2009.

[139] V. V. Lukin, M.S. Zriakhov, N.N. Ponomarenko, S.S. Krivenko and Z.J. Miao. “Lossy compression of images without visible distortions and its application”. IEEE 10th Inter. Conf. Signal Proc. (ICSP), Beijing, China, Oct. 24-28 2010, pp. 698- 701.

[140] S. Chuah, S. Dumitrescu, X. Wu, “*l2* Optimized Predictive Image Coding with $l\infty$ Bound”, IEEE Trans. Image Processing, vol . 22, no. 12, pp. 5271 – 5281, Dec. 2013.

[141] O.-J. Kwon and R. Chellappa. (1998 May) “Region adaptive subband image coding”. IEEE Trans. Image Proc., 7(5), pp. 632-648.

[142] J. W. Woods and S. D. O’Neil. (1986). “Subband coding of images”. IEEE Trans. Acoust., Speech, Signal Proc., vol. 34, pp.1278–1288.

[143] S. E. Reichenbach, Z.-U. Rahman and R. Narayanswamy.(1993, Sep.) "Transform Coding Image Compression for Information Efficiency and Restoration". Jour. of Visual Comm. & Image Repre. 4(3), pp. 215–224.

[144] V.K. Goyal. (2001 Sep.) "Theoretical foundations of transform coding". IEEE Signal Proc. Magazine, 18(5), pp. 9-21.

[145] M. Xu, A. Kuh. (1996 Jul.). "Image Coding Using Feature Map Finite-State Vector Quantization". IEEE Signal Proc. Lett. 3(7), pp. 215-217.

[146] A. Gersho, R. M. Gray. Vector Quantization and Signal Compression. Kluwer Academic Publishers Norwell, MA, USA, 1991.

[147] N. Phamdo, N. Farvardin, T. Moriya. (1993 May). "A Unified Approach to Tree-Structured and Multi-Stage Vector Quantization for Noisy Channels". IEEE Trans. IT, 39(3), pp. 835-850.

[148] Z. X. Xiong, K. Ramchandran, M. T. Orchard, and Y.-Q. Zhang. (1999 Aug.). "A Comparative Study of DCT- and Wavelet-Based Image Coding". IEEE Trans. Circ. & Sys. for Video Tech. 9(5), pp. 692-695.

[149] J. R. Wang and R. Liu. "Low complexity DCT-based distributed source coding for hyperspectral image". The Fourth International Conference on Communications and Networking, Xian China, Aug. 2009, pp. 1-5.

[150] K. H. Talukder, K. Harada (2007). "Haar Wavelet Based Approach for Image Compression and Quality Assessment of Compressed Image" IAENG Inter. Jour. of Applied Mathematics, 36(1). http://www.iaeng.org/IJAM/issues_v36/issue_1/IJAM_36_1_9.pdf

[151] S.G. Mallet, (1989 Jul.). “A Theory for Multiresolution Signal Decomposition: The Wavelet Representation”. IEEE Trans. PAMI, 11(7), pp. 674-693.

[152] M. Vetterli. (Sept. 2001). “Wavelets, approximation, and compression”. IEEE Signal Processing Mag., 18(5), pp. 59-73.

[153] B.E. Usevitch. (2001 Sep.). “A tutorial on modern lossy wavelet image compression: foundations of JPEG 2000.” IEEE Signal Proc. Magazine, 18(5), pp. 22-35.

[154] Akansu, Ali N.; Haddad, Richard A. (1992), Multiresolution signal decomposition: transforms, subbands, and wavelets. Boston, MA: Academic Press.

[155] F. Pan, X. Lin, S. Rahardja, W.S. Lin, E.P. Ong, S.S. Yao, Z.K. Lu and X.K. Yang .(2004 Jul.). “A locally adaptive algorithm for measuring blocking artifacts in images and videos”. Signal Processing: Image Communication, 19(6), pp. 499-506.

[156] P. N. Huu, V. Tran-Quang, T. Miyoshi. “Image compression algorithm considering energy balance on wireless sensor networks”. The 8th IEEE International Conference on Industrial Informatics (INDIN), Osaka, Japan, Jul. 2010, pp. 1005 – 1010.

[157] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K..F. Kelly and R.G. Baraniuk. (2008 Mar.) “Single-pixel imaging via compressive sampling”. IEEE Signal Proc. Magazine, 25 (2), pp. 83-91.

[158] R. Yang, L. Yin, M. Gabbouj, J. Astola, Y. Neuvo, “Optimal weighted median filters under structural constraints”. IEEE Trans. Signal Processing, vol. 43, pp.591–604, Mar. 1995

[159] R. C. Hardie and K. E. Barner, “Rank conditioned rank selection filters for signal restoration”. IEEE Trans. Image Processing, vol. 3, pp.192–206, Mar. 1994.

[160] A. Ben Hamza, P. Luque, J. Martinez, and R. Roman, “Removing noise and preserving details with relaxed median filters”. J. Math. Imag. Vision, vol. 11, no. 2, pp. 161–177, Oct. 1999.

[161] A.K.Jain, Fundamentals of digital image processing. Prentice-Hall,1989

[162] P. Salembier, “Overview of the MPEG-7 standard and of future challenges for visual information analysis,” EURASIP J. Appl. Signal Process., vol. 4, pp. 1-11, April 2002.

[163] Hai Tao ; Moorhead, R.J. “Progressive transmission of scientific data using biorthogonal wavelet transform” IEEE Conf. Visualization, Washington, DC, Oct 1994, pp. 93-99.

[164] Hai Tao; Moorhead, R.J. “Lossless progressive transmission of scientific data using biorthogonal wavelet transform” Proceedings of IEEE International Conference Image Processing vol.3 pp. 373- 377, 1994.

[165] R. J. Hyndman, A. B. Koehler (2006). "Another look at measures of forecast accuracy". International Journal of Forecasting 22(4): 679–688.

[166] Huynh-Thu, Q.; Ghanbari, M. (2008). "Scope of validity of PSNR in image/video quality assessment". Electronics Letters 44 (13): 800-801.

[167] Solomon, C.J., Breckon, T.P. Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab (1st). Wiley, 2011

[168] Maria Petrou. Costas Petrou. Image Processing: The Fundamentals (2nd). Wiley. 2010

[169] R. Fisher, K. Dawson-Howe, A. W. Fitzgibbon, C. Robertson, E. Trucco; Dictionary of Computer Vision and Image Processing, John Wiley and Sons, 2005.

[170] Burt, P.J and Adelson, E.H. (1983). "The Laplacian Pyramid as a Compact Image Code", IEEE Trans. Communications, 31 (4): 532–540.